A NOVEL HYBRID AGGREGATED CLASSIFIER FOR
INTERNET TRAFFIC CLASSIFICATION

G. Rubadevi 1, R. Amsaveni 2

1 Research scholar, Department of Computer Science
2 Asst. Professor, Department of Information Technology, PSGR. Krishnammal College for Women, India

ABSTRACT:
The classification and identification of network application from network traffic flow provides various advantages to a number of fields such as security monitoring, intrusion detection and to tackle a number of network security problems including lawful interception. In this paper traffic flow is described by using the discretized statistical features. The flow correlation information of the network traffic flow is modeled by Flow Container (FC). In this paper novel hybrid aggregated classifier is proposed. First, low density flow and high density flow is analyzed. For Low density flow C4.5 classifier is used and high density flow Naïve Bayesian classifier is used and finally aggregated result is provided. The aggregated result is compared with machine learning algorithm such as Single Naïve Bayesian predictor. The proposed system enhances the accuracy rate as well as improves the performance of the system.

Keywords: Traffic classification, Flow Container Construction, Machine learning algorithms, Single NB Predictor, Hybrid Aggregated Classification.

[1] INTRODUCTION

Traffic classification is an automated procedure which classifies computer network traffic according to various constraints into a number of traffic classes. Application related traffic classification is basic technology for recent network security. The traffic classification can be used to find out the worm propagation, intrusions, and patterns indicative of denial of service attacks, and spam spread. Internet traffic is mainly occurs due to File Sharing, Streaming Media, Videoconferencing, Malware etc. The development of network technology and application leads to severe shortage of the network resource. Identifying and control of the internet traffic flows with high efficiency is the key problem to solve.

Traditional method of traffic classification techniques may focused on Port Based application Deduction, Packet based analysis, Payload based application deduction and the modern techniques used to classify the internet traffic is statistics based classification, Flow-based Classification.
[1.1] TRADITIONAL METHOD

Port-Based Application Deduction: Application is combining with a port number, the common network traffic classification method is based on port number and it uses only packet header. In this method the classification is based on well-known port number to a given traffic type, for example web traffic is associated with TCP port 80. The port number of any application is not default, which leads to failure of the port based method.

Packet-Based Analysis: The packet based analysis will give most accurate solution but, there are many pitfalls with this method. Some protocols are encrypted, thus their data stream is intentionally hidden from sniffer programs. Another problem is that, there is no protocol description for organization protocols. Parsing all protocols for all users separately is a computationally unavailable and risky task. In this method, parsing the payload of network packets is highly privacy violation. Signature method of traffic classification consumes less resource which is searching for specific byte patterns, even when packet s in the stateless manner. The signature based methods is very difficult to maintain signatures with maximum ratio of true positives and minimum ratio of false positives.

[1.2] RECENT METHOD

Statistics based classification: Packet level trace generates n number of zero payload flows where peer try to connect each other. In this case some statistical feature of the packet-level trace is grabbed and used to classify the network traffic. This approach is feasible to determine the application type, but specific application/client cannot be determined in general. These flow characteristics can be highly coded manually or another way is to automatically extract the features of a particular kind of traffic. This method is achieved by combining statistical method with artificial intelligence. There is various data mining approaches combination to apply statistical based classification. Applying statistical based classification will give high accuracy for traffic classification, but the result cannot be exact and accept minor classification errors.

Flow-based Classification: Traffic application based on flow-level data with the same and high level of accuracy is very difficult, because it consist of less detailed input. For application behavior, analyzing the application constraints makes the classification more feasible. The connection patterns is the novel approach to classify traffic based on the application groups, It is represented by graphs, where nodes provides ip address and port pairs information and edge represents flows between source and destination nodes. Connection patterns are analyzed at three levels of details, the social, the functional and the application level. This method operates within the information having no access to payload information, no knowledge about port number and no information behind what current flow collectors give. On the other hand, connection patterns require a high quality of flow information and finished flow period to perform the analyses.

The proposed work of this paper uses the recent method of traffic classification such as Flow based traffic classification and Statistics based classification.
[2] PROPOSED CLASSIFICATION SCHEME

[2.1] PREPROCESSING

The preprocessing is the first process; the system captures internet protocol packets crossing a targeted network and build traffic flows by analyzing the IP packet header. A traffic flow contains IP packets with the 5-tuple such as source IP, source port, destination IP, destination port, and transport layer protocol.

[2.2] FLOW CONTAINER CONSTRUCTION

The correlated flows are determined by using heuristic way and model them using flow container. The correlated flows means, the flows represent same destination IP, destination port and transport layer protocol in a certain period of time span and form a Flow Container. Such correlation information can be utilized to improve the classification results. Therefore, we aim to aggregate the individual predictions of the correlated flows so as to conduct more accurate classification.

The architecture diagram given below shows the proposed classification using hybrid Aggregated classifier.
[2.3] FEATURE EXTRACTION AND FEATURE DISCRETIZATION

There are number of feature available for internet traffic classification traffic classification. Some of the feature used in previous method of traffic classification includes

1. Packet length statistics, window size, byte ratio of received packet, number of packets, number of packet per flow, transport protocol. [3].
3. 249 features including port, flow duration and inter arrival length and packet length. [8]
4. Flow duration statistics, data volume, number of per flow. [3]
5. Number of packets, packet length and statistics of inter arrival, duration flow. [6]
6. Mean, variance, $1^{st}$ and $3^{rd}$ quartiles, median, minimum, maximum of TCP/IP size. [7]
7. Mean, Variance, $1^{st}$ and $3^{rd}$ quartiles, minimum, maximum and SD of payload size. [10]
8. Effective Bandwidth based upon entropy. [8]

In proposed method of traffic classification, feature is extracted and discretized to represent traffic flow. Unidirectional Statistical Feature are used for classification to improve the accuracy.

[2.4] UNIDIRECTIONAL STATISTICAL FEATURE EXTRACTION

<table>
<thead>
<tr>
<th>S.NO</th>
<th>Types of Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Flow Packets transferred in unidirection.</td>
</tr>
<tr>
<td>2</td>
<td>Length of the packets volume of byte transferred in unidirectional flow.</td>
</tr>
<tr>
<td>3</td>
<td>Minimum value of Packet size</td>
</tr>
<tr>
<td>4</td>
<td>Maximum value of Packet size</td>
</tr>
<tr>
<td>5</td>
<td>Mean of the Packet size</td>
</tr>
<tr>
<td>6</td>
<td>Standard deviation of the Packet size</td>
</tr>
<tr>
<td>7</td>
<td>Variance of the Packet size</td>
</tr>
<tr>
<td>8</td>
<td>Minimum of Inter packet time in unidirectional flow</td>
</tr>
<tr>
<td>9</td>
<td>Maximum of Inter packet time in unidirectional flow</td>
</tr>
<tr>
<td>10</td>
<td>Mean of the Inter packet time in unidirectional flow</td>
</tr>
<tr>
<td>11</td>
<td>Variance of Inter packet time in unidirectional flow</td>
</tr>
<tr>
<td>12</td>
<td>Standard deviation of Inter packet time in unidirectional flow</td>
</tr>
</tbody>
</table>
[3] FEATURE SELECTION ALGORITHMS

[3.1] CORRELATION-BASED FEATURE SELECTION (CFS)

CFS estimates and ranks the subset of features quite than individual features. It chooses the set of attributes which are highly associated with the class in addition those attributes are low intercorrelation. With CFS several heuristic searching approaches such as hill climbing and best first are often functional to search the feature subsets space in reasonable time. CFS first computes the feature-class matrix and feature to feature correlations from the training data after that searches the feature subset space using a best first.

\[ M_s = \frac{k \bar{r}_{cf}}{\sqrt{k + k(k - 1)\bar{r}_{ff}}} \]

Where \( M_s \) is the correlation between the summed feature subset \( S \), \( k \) is the number of subset feature, \( \bar{r}_{cf} \) is the average of the correlation between the subsets feature and the class variable, and \( \bar{r}_{ff} \) is the average inter-correlation between subset features.

[3.2] FEATURE DISCRETIZATION

In machine learning, discretization refers to the method of converting numeric feature values into a number of intervals, and then mapping each interval to a discrete nominal symbol. These discrete nominal symbols are used as new values, replacing the original numeric values of the features. Discretization is defined as follows:

Discretization is a function \( Q : D \rightarrow C \) Assigning a class \( c \in C \) to each value \( v \in D \) in the domain of the attribute being discretized. A desirable feature for a practical discretization is: \(|D| > |C|\) i.e., a discretized feature has fewer possible values than the original not-discretized one.

The choice of the intervals is the key process in feature discretization. There are two basic approaches to the problem of discretization: One is to quantize each feature in the absence of any knowledge of the classes of the instances in the training set - so-called unsupervised discretization. The other is to take the classes into account when discretizing – supervised discretization.

[3.3] MACHINE LEARNING

We use machine learning algorithms to plot the occurrences of network traffic flows into various network traffic classes. Each flow is described by a set of unidirectional statistical features and respected feature values. The brief statistical feature is a calculated from one or more packets includes mean, maximum, minimum, standard deviation, variance of packet length, inter arrival times, number of packet, number of packets in unidirectional flow. Each traffic flow is differentiated by the same set of features, though each will reveal in different feature values based on the network traffic class to which it belongs. Machine learning algorithms that have been used for internet traffic classification usually come into the categories of being supervised or unsupervised classification. For supervised algorithms the
category of each correlated flow must be known before learning. A classification model is built using training network traffic. This developed model is able to predict class belongings for new vector by analyzing the feature values of unknown traffic flows.

In this paper, the low density and high density of the traffic flow is calculated. Low density traffic flow is classified by C4.5 classifier and high density flow is classified by Naive Bayesian classifier. Then aggregated result is calculated and provided as classification.

### [3.4] HIGH DENSITY – NB CLASSIFIER

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. Assumes an underlying probabilistic model and it allows us to capture uncertainty about the model in a principled way by determining probabilities of the outcomes. Bayesian classification provides practical learning algorithms and prior knowledge and observed data can be combined. Bayesian. It calculates explicit probabilities for hypothesis and it is robust to noise in input data.

The Naive Bayesian classifier is based on Bayes’ theorem with independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large traffic datasets. Bayes theorem provides a way of calculating the posterior probability, \( P(c|x) \), from \( P(c) \), \( P(x) \), and \( P(x|c) \). This assumption is called class conditional independence.

\[
P(c|x) = \frac{P(x|c)P(c)}{P(x)}
\]

- \( P(c|x) \) is the posterior probability of class (target) given predictor (attribute).
- \( P(c) \) is the prior probability of class.
- \( P(x|c) \) is the likelihood which is the probability of predictor given class.
- \( P(x) \) is the prior probability of predictor.

### [3.5] LOW DENSITY – C4.5 CLASSIFIER

The C4.5 classifier uses an extension of information gain known as gain ratio. Let \( S \) be set consisting of \( s \) data samples with \( m \) distinct classes. The expected information needed to classify a given sample is given by

\[
I(s) = - \sum_{i=1}^{m} p_i \log_2 p(i)
\]

where \( p_i \) is the probability that an arbitrary sample belongs to class \( Ci \) and is estimated by \( \frac{s_i}{s} \). Let attribute \( A \) has \( v \) distinct values. Let \( sij \) be number of samples of class \( Ci \) in a subset \( Sj \). \( Sj \) contains those samples in \( S \) that have value \( aj \) of \( A \). The entropy, or expected information based on the partitioning into subsets by \( A \), is given by

\[
E(A) = - \sum_{i=1}^{M} I(S) \frac{s_{1i} + s_{2i} + \cdots + s_{mi}}{s}
\]
The encoding information that would be gained by branching on \( A \) is

\[
\text{Gain (A)} = I(S) - E(A)
\]

C4.5 uses gain ratio which applies normalization to information gain using a value defined as

\[
\text{SplitInfo}_A(S) = - \sum_{i=1}^{v} \left( \frac{|S_i|}{|S|} \right) \log_2 \left( \frac{|S_i|}{|S|} \right)
\]

The above value represents the information generated by splitting the training data set \( S \) into \( v \) partitions corresponding to \( v \) outcomes of a test on the attribute \( A \).

The gain ratio is defined as

\[
\text{Gain Ratio (A)} = \frac{\text{Gain (A)}}{\text{SplitInfo}_A(S)}
\]

The attribute with the highest gain ratio is selected as the splitting attribute. The non leaf nodes of the decision tree generated are considered as relevant attributes.

The summary of decision tree algorithm is given:

i. Choose an attribute that best differentiates the output attribute values.
ii. Create a separate tree branch for each value of the chosen attribute.
iii. Divide the instances into subgroups so as to reflect the attribute values of the chosen node.
iv. For each subgroup, terminate the attribute selection process if:
   (a) The members of a subgroup have the same value for the output attribute, terminate the attribute selection process for the current path and label the branch on the current path with the specified value.
   (b) The subgroup contains a single node or no further distinguishing attributes can be determined. As in (a), label the branch with the output value seen by the majority of remaining instances.
v. For each subgroup created in (iii) that has not been labeled as terminal, repeat the above process.

[4] EXPERIMENTAL RESULT AND EVALUATION

[4.1] DATASET

In the experiments, we use the full packet traffic trace which is collected from educational campus. The traffic flow is captured by using well known network traffic monitoring tool WIRESHARK. There are 2330 flow are captured for 1 minute period of time. 80% of the flow traffic packet is used for training and 20% of flow traffic packet is used for testing data.
[4.2] EXPERIMENTAL SETUP

MATLAB 2012 is used to carry out the experiment. MATLAB provides an intuitive language and a flexible environment for technical computations which integrates mathematical computing and visualization tools for data analysis and development of algorithms and applications. The machine learning algorithm such as single NB classifier and hybrid aggregated NB classifier and C4.5 classifier are used for classification. There are seven class is classified which includes ALC, ARP, UDP, CDP, DHCP, LLMNR, NBNS.

[4.3] ACCURACY FOR DIFFERENT APPLICATION PROTOCOL

<table>
<thead>
<tr>
<th>S.NO</th>
<th>APPLICATION PROTOCOL</th>
<th>SINGLE NB CLASSIFIER</th>
<th>HYBRID AGGREGATED CLASSIFIER(C4.5 AND NB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ALC</td>
<td>71.42</td>
<td>85.7</td>
</tr>
<tr>
<td>2</td>
<td>ARP</td>
<td>79.16</td>
<td>87.5</td>
</tr>
<tr>
<td>3</td>
<td>UDP</td>
<td>81.81</td>
<td>90.91</td>
</tr>
<tr>
<td>4</td>
<td>CDP</td>
<td>79.16</td>
<td>87.50</td>
</tr>
<tr>
<td>5</td>
<td>DHCP</td>
<td>79.15</td>
<td>91.66</td>
</tr>
<tr>
<td>6</td>
<td>LLMNR</td>
<td>83.33</td>
<td>91.67</td>
</tr>
<tr>
<td>7</td>
<td>NBNS</td>
<td>76</td>
<td>92</td>
</tr>
</tbody>
</table>

[4.4] EVALUATION RESULT

Accuracy: The number of correctly classified instances over the total number of instances.
Precision: Precision value is calculated is based on the retrieval of information at true positive prediction, false positive. The number of correctly classified instances of class Y over the total number of instances classified as belonging to class Y.
Recall: Recall value is calculated is based on the retrieval of information at true positive prediction, false negative, the number of correctly classified instances of class Y over the total number of instances of class Y.
F-measure comparison: F-measure distinguishes the correct classification of document labels within different classes. For the overall performance metrics of the classifier, weighted average of particular performance metrics are used. It is defined as follows:

\[
F\text{-measure} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
\]
[4.5] OVERALL PERFORMANCE METRICS OF THE CLASSIFIER

![Graph showing performance metrics]

[5] CONCLUSION

The existing system has several drawbacks such that doesn’t analyses the density of the data. For high density and also low density, this system used the same classifier. This will degrades the performance of the system and also less number of features is extracted from the data. This will degrade the accuracy rate of the system. With the intension of overcome these problems as well as to increase the accuracy rate of traffic classification, we are proposing the novel hybrid aggregated classifier. Our proposed novel hybrid aggregated classifier contains the Naïve Bayesian classifier and C4.5 classifier. Based on the density of the data, in this system uses these two classifiers for traffic classification purpose. In addition, proposed system is extracts more features from the traffic data in order to enhance the accuracy rate as well as improve the performance of the system. The experimental results show that the proposed scheme can achieve much better classification performance than existing internet traffic classification methods.
REFERENCES:


[15] [Online]. Available https://www.wireshark.co.in