Generic Attributes for Skype Identification Using Machine Learning

Rozanna Nadeera Jesudasan[1], Philip Branch, Jason But
Centre for Advanced Internet Architectures. Technical Report 100820A
Swinburne University of Technology
Melbourne, Australia

***Abstract*-Identification of Skype traffic using machine learning is an area of current research interest. Previous Skype classifiers have usually been reliant on version specific features. Consequently, a classifier that works for a specific version of Skype is unlikely to work for successive versions. Classification of Skype has been successful with previous research showing 98% precision for Skype version 3. But classification using Skype version 3 attributes were not successful in identifying Skype version 4. In our experiments, we use machine learning to classify Skype version 3 and 4 for a subflow size of 100 with characteristics common to both versions. We discuss attributes that are generic to Skype and show that Skype can be identified with 97% precision and 86% recall.**

Keywords- Machine Learning ,UDP Traffic, Classification,

I. Introduction

Realtime classification of IP traffic flows is an increasingly important area with applications in lawful interception of calls, intrusion prevention, QoS management and market research[1]. In the past classification has largely been based on well-known port numbers or Deep Packet Inspection. Unfortunately, the use of dynamic port numbers and encryption has made these approaches much less effective than in the past.

Consequently, Machine learning has attracted attention as an alternative technique [1][2]. Machine learning involves the construction of a classifier that uses characteristics of the flow to identify its class. Most interest and success has been in using statistical characteristics of sub-flows of packets, such as mean packet length, inter-arrival times and characteristic packet lengths [1][2]. These techniques have been used successfully to identify Skype traffic and BitTorrent traffic[1][2] with precisions of around 98%.

Although very successful in identifying traffic classes for for specific versions of the application, the technique is less successful at identifying different versions of the same application. The problems arise from the fact that applications change attributes from one version to the next. As an example, Skype traffic, uses different codecs for versions 3 and 4 resulting in a change of characteristic packet lengths. This change affects the classification of Skype version 4 when the characteristic packet lengths from Skype version 3 are used as attributes.

The experiments we have done are aimed at identifying characteristics of Skype that are common to versions 2, 3 and 4. We carried out the experiments by first capturing the packet flows and dividing them into sub flows. We then calculated the statistical characteristics for each sub-flow. These statistics are then used to plot CDF graphs to decide which attributes could be used for classification. The selected attributes are then used in a classification tool WEKA to test if Skype can be classified accurately [6].

Our paper is organised as follows. Section II provides information regarding Skype traffic. Section III gives an introduction to machine learning techniques used in our experiments. Section IV gives details on selected attributes used for traffic classification. Section IV also shows the characteristics and classification of Gtalk with Games and Other traffic. Section V provides the results obtained and and Section VI gives the conclusion.

II. Skype

Skype is an application that can be used to make voice calls over the Internet. Skype version 3 used the SVOPC (Sinusoidal Voice Over Packet Coder), which is a variable bit rate codec that changes with the available bandwidth [1]. Skype version 4 uses a codec called SILK, the details which have been made freely available by Skype unlike the SVOPC codec used previously. The SILK codec provides high bit rates to improve the audio quality. It encodes frames every 20ms and can combine up to 5 frames per packet. This allows Skype version 4 to create packets at intervals of 20,40,60,80 or 100 ms [3][4].

Skype is a peer-to-peer VOIP application which uses nodes which have enough CPU, memory and bandwidth to connect Skype clients. Each Skype client has a list of nodes or Super nodes. A super node is usually a network that containing a large amount of resources such as an ISP or any large network. A node can be any resource such as a router. Skype has the ability to pass the NAT and firewalls by using UDP or TCP as a transport protocol making it even harder to identify Skype[5].

Skype is also difficult to detect because it is encrypted and uses dynamic port numbers. Because of this, methods such as deep packet inspection and port number classification are unable to identify Skype

[1] This author is currently an engineering student at Swinburne University of Technology. This report was written during the author's winter internship at CAIA in 2010.

accurately. Traffic classification using machine learning has been successful with Skype version 3.2 [1]. But with the release of Skype version 4, it has been difficult to identify the newer version with the classifiers trained using the older versions characteristics. Hence we concentrate on identifying characteristics common to both versions 3 and 4 of Skype.

III. Machine Learning

Machine learning is a method of constructing a classifier. We use machine learning techniques to classify IP traffic flows based on a given set of data [8] [10]. The characteristics of traffic types are given to the classifier in the form of statistical quantities such as mean, variance and standard deviation. A machine learning algorithm is used to train a classifier to recognise the traffic class based on the given data. This method has previously been used to classify Skype and BitTorrent traffic [1][2].

A. Subflow

For a classifier to be useful in a realtime environment, it needs to be timely. Consequently, analysing the full flow of Skype traffic is not a suitable solution. As a result we use the method of training and classifying on sub flows where only a part of the flow (typically a few hundred packets) is used [6]. This method has also been used in classification of BitTorrent traffic[2]. Sub flows are created for a full flow by segmenting it into sub flows of fixed number of packets. The advantage of using sub-flows is not only limited to fast classification, but also the ability to correctly classify a class with subsequent sub-flows even if one sub-flow might not be classified correctly.

B. Classification

Identification of characteristics is done by capturing traffic flows and observing the characteristics of the partial or full flows. In our experiments we used partial flows to obtain the statistical characteristics of traffic types. For each characteristic graphs are plotted to identify suitable statistics that are able to separate Skype traffic from the other traffic types. These statistics are then used with a classifier to separate Skype from the other traffic types.

C. The Classifier

Two algorithms J48 and Naive Bayes were used to classify both version 3 and 4 of Skype traffic using WEKA. WEKA is a tool that has machine learning algorithms for classification of data [7]. The J48 classifier is a decision tree supervised classifier which branches out its selections based on the statistical characteristics.

The Naive Bayes classifier is also a supervised classifier for which the the input is given along with examples of the features of the output [7].

```
J48 pruned tree
Mean <= 26.5
| DAbsAutoCorr <= 0.157061
| | DAbsAutoCorr <= 0.010246
| | | IndDisp <= 0: Other (6.0/1.0)
| | | IndDisp > 0: Skype (5.0)
| | DAbsAutoCorr > 0.010246: Skype (256.0/2.0)
| DAbsAutoCorr > 0.157061: Other (23.0)
Mean > 26.5: Other (2892.0/19.0)
```

Figure 1: Example decision tree using WEKA

Each of these algorithms were tested using ten fold cross-validation. The cross-validation method divides the given data in to ten samples. One sample is used for testing and the other 9 samples are used for training. This is done ten times with each sample being tested once. The final output gives an average of the results for each training and testing sequence. Classifiers that were successful in this test were then tested on different Skype versions.

To measure the effectiveness of the classification we consider a few statistics such as Recall and Precision. Recall is the fraction of samples that are correctly classified for a class from the total within that class. The precision is the fraction of samples that have been correctly classified into a particular class out of all the classes by considering the total number of classifications within that class[3][4]. The greater the precision and the recall the greater the effectiveness of the classifier.

IV. Traffic Classification

A. Tools and Data

Tcpdump was used to capture data for Skype versions 3 and 4.The Other pcap files were obtained from University of Twente[9]. Each of the pcap files was used to obtain the flows and create sub-flows for each class of traffic. Next the characteristics were calculated for each sub-flow size. This was then written to a .CSV file. The .CSV file was used to create an .ARFF file, which is the WEKA input file. The .ARFF file has attributes, as well as the statistical data for each attribute type. We created the WEKA input files for all the attribute combinations for a sub-flow size of 100 for initial testing. A sub-flow size of 100 was appropriate because it gave many data sets to analyse for each traffic type.

B. Attributes

The attributes that we tested were

- Mean of packet length,
- Variance of packet length,
- Index of Dispersion which is the variance of packet lengths divided by the mean packet length,
- Two packet difference which is the difference in packet lengths between two consecutive packets, the

- Absolute two packet difference which is the absolute difference in packet lengths between two consecutive packets,
- Ratio of packets between the forward and reverse flows and
- Inter-arrival time.

C. Mean Packet Length

We obtained the mean lengths of packets per sub-flow for Skype versions 2,3 and 4. We also Obtained the mean of the Other traffic, Gtalk and Games (for 2 ,4 and 9 players).

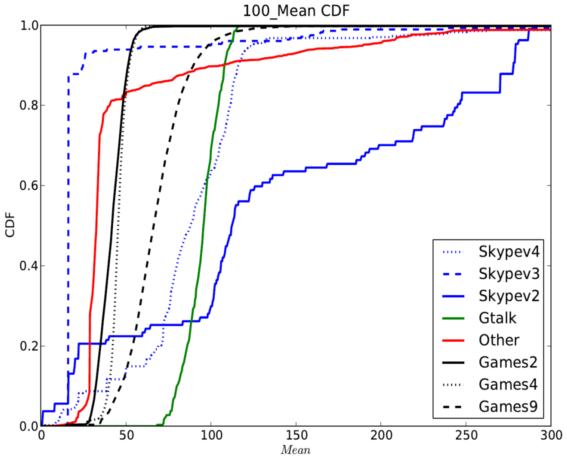

Figure 2: Mean packet lengths length for sub-flow size 100

Most of the mean packet lengths for Skype version 2 is less than 250. For Skype version 3 most of the packet lengths are less than 50, with Skype version 4 having an average packet length less than 150 for most packets. For Games we observed that as the number of players increased the average packets increased. We also observed that for 80% of the Other traffic the mean was less than 50.

D. Variance of Packet Length

The variance of the packet length gives a measure of how packet lengths vary within each sub-flow.

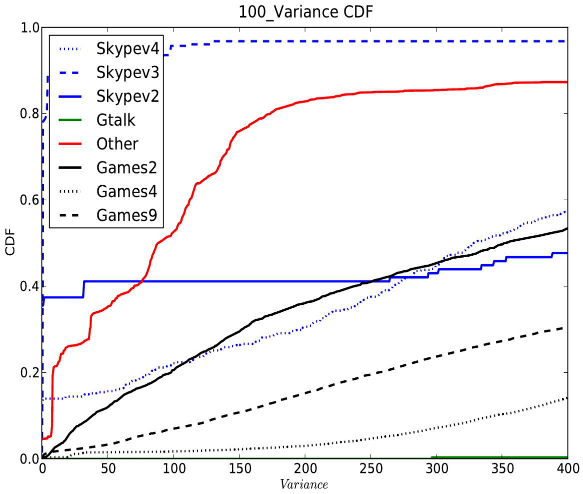

Figure 3: Variance of packet lengths for sub-flow size 100

We see that Skype versions 2 and 4 have a high variance compared with Skype version 3. Skype version 3 has the smallest variance in packet lengths. The variance seen for Games traffic increases approximately linearly as the number of players increase. Since the variance of all traffic types are spread across a large range we decided that the variance was not a suitable characteristic to be used for classification. This was proven during the testing with WEKA. We observed that the variance could not be used to classify both the Skype versions accurately.

E. Index of Dispersion

We also obtained the index of dispersion which is the variance in packet lengths compared to the mean packet length. The CDF of the index of dispersion is shown below.

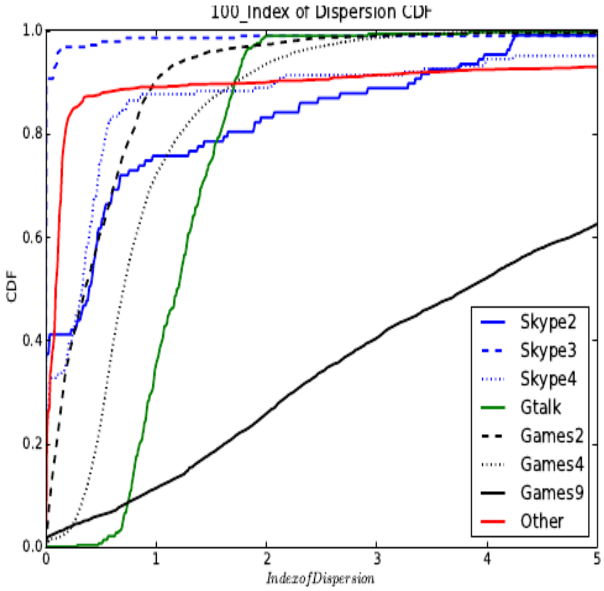

Figure 4: Index of Dispersion for sub -flow size 100

From the graph we see that all traffic types have an index of dispersion that is less than 5, for more than 80% of the sub-flows. We also observed that for all the three Skype versions the index of dispersion was different. Skype version 3 had the least dispersion and Skype version 4 had a larger index of dispersion compared to Skype version 3. But for all three versions of Skype the index of dispersion does did not vary greatly. So the index of dispersion is a good attribute to classify Skype traffic. The test results with WEKA showd that the index of dispersion was needed to classify Skype.

For Games traffic, as the number of players increased the index of dispersion increased. This is due to the mean and the variance increasing linearly as the number of players increase[10].

F. Ratio

The ratio is calculated by dividing the packet lengths in the forward and reverse flow directions for each sub-flow. We normalize the ratio by dividing each small packet with its respective large packet for the unidirectional flow to obtain a value between 0 and 1.

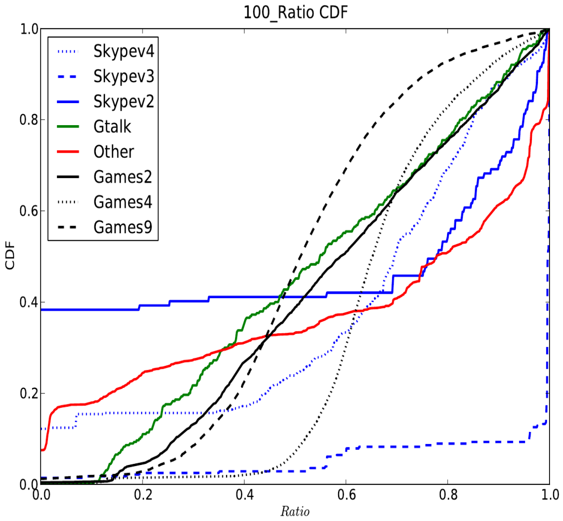

Figure 5: Ratio of packet lengths for sub-flow size 100

The ratio of the packet length for Skype version 3 is ~1 for most of the packets. Hence the packets in the forward and reverse direction have the same packet lengths. But this is not true for Skype version 2 and 4. As seen from the graph, Skype version 2 has 80% of its traffic with a ratio less than 0.9. Similarly for Skype version 4, 80% of the traffic has a ratio less than 0.8. The plots for the Other traffic and Games traffic shows that the packet ratios are not always equal in both directions of the flow.

G. Two Packet Difference

The two packet difference is used to observe the self similarity of a certain type of traffic. It is calculated by obtaining the difference in packet lengths between two consecutive packets. The CDF of the two packet difference is shown below.

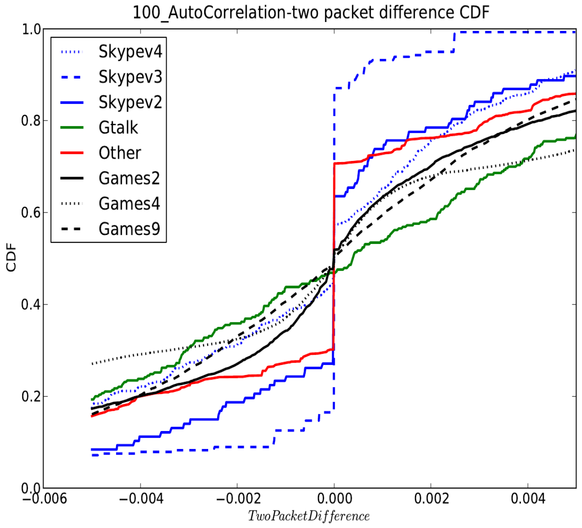

Figure6 :Two Packet Difference for sub-flow size 100

The two packet different had a slightly better separation in terms of identifying Skype and Other traffic. Even though the separation for each of the traffic types were small, it gives a good statistical seperation for Skype version 3. As we can see, Skype version 3 has a zero two packet difference due to it having equal packet lengths 85% of the time. But this is not true for Skype versions 2 and 4. Skype version 4 does not have a zero two packet difference for a most of its packets. This characteristic could also be seen for Gtalk and Games. For the Other traffic, we observe that around 40% to 60% of the packets have a two packet difference of zero. This is due to the Other traffic having DNS queries within the capture. Because the size of these DNS queries are approximately equal in length it gives a two packet difference of zero.

H. Absolute Two Packet Difference

The absolute two packet difference is the two packet difference by adding the absolute value of the difference of two consecutive packets for each sub-flow.

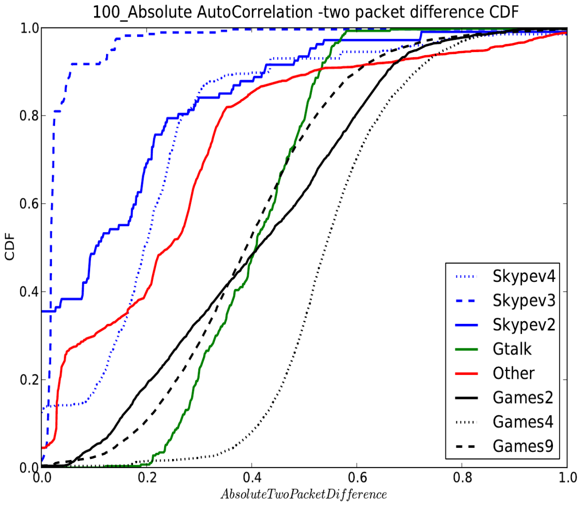

Figure 6:Absolute two packet difference for sub-flow size 100

This graph shows that majority of the Skype traffic has an absolute two packet difference that is less than 0.6. The Other traffic has most of the packets with an absolute two packet difference of less than 0.6. For Games we observed that almost 100% of the packets have an absolute two packet difference less than 0.6. We expected the absolute two packet difference to be a useful attribute in identifying Skype traffic against the Other traffic, but after classification the absolute two packet difference did not contribute a to successfully classifySkype traffic.

I. Inter-arrival time

The inter-arrival time is plotted by finding out the average inter-arrival time for each sub-flow. The CDF graph for inter-arrival time is shown below.

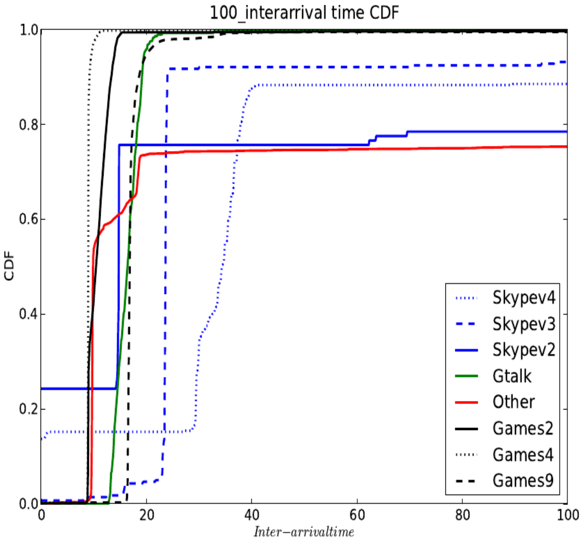

Figure 7: Inter-arrival time for sub-flow size 100

Games and Gtalk both had an inter-arrival time that is less than 20ms. Similarly 80% of the inter-arrival times for Skype version 3 and 4 were less than 40ms. For the Other traffic and Skype 2 traffic the inter-arrival time takes more than 100ms to reach the 80% in the CDF graph. Because of these characteristic Skype version 3 and 4 could be separated from the Other traffic, during our experiment we found that the inter-arrival time was needed along with other attributes to classify Skype traffic.

V. Classifier Results

A. Training and Testing

Each input file combination was tested and trained using the WEKA classifier. The ten fold cross-validation was used for testing and training the given data. Initially the training was done for Skype version 3 and the Other traffic. Next the testing was done for Skype version 4 and Other traffic. The J48 classification was used for both training and testing during this experiment. We trained the data for combinations from one to seven attributes and tested with all combinations.

B. Generic attributes

For many attributes the training results with the J48 classifier showed a better classification of Skype version 3 with the Other traffic. But when testing with Skype version 4 most of Skype was classified as Other. But we found some attributes gave positive classification results in classifying Skype version 3 and 4. Initially the attribute combinations of Mean, Index of Dispersion and Inter-arrival time gave a high precision. But by including the additional attribute Two Packet difference there was an increase in the recall and the precision.

C. WEKA Results

The results shown below are using the four generic attributes that classify Skype versions 3 and 4.

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.858	0.004	0.974	0.858	0.912	Skype
0.996	0.142	0.979	0.996	0.987	Other

Table 1: WEKA-Statistical results for Skype

```
=== Confusion Matrix ===

  a     b   <-- classified as

484    80 |   a = Skype

 13  3665 |   b = Other
```

Figure 8: Confusion Matrix -WEKA

The confusion Matrix gives a detailed explanation of the classification of both Skype and Other traffic. Out of 564 Skype sub-flows, 484 sub-flows are correctly classified as Skype and 80 sub-flows are incorrectly classified as Other. Similarly for the Other traffic 3665 are correctly classified.

As shown in table 1, the precision of 97% for Skype and ~98% for the Other traffic was obtained by using the four attributes Mean, Index of Dispersion, Inter-arrival time and Two Packet difference. And the results show that it is possible to classify Skype successfully with at least a precision of 97% and a recall of 86%.

D. Improvements

As seen from the results in table 1 and figure 8, Skype traffic has a smaller number of sub-flows compared to the Other traffic. To increase the validity of the results it is better to classify approximately equal numbers of sub-flows for both Skype and the Other traffic. We expect this change to give a improved precision for the classification of Skype traffic.

VI. GTALK and Games

We also wanted to classify Gtalk traffic with Other traffic as well as Games traffic. Mostly the characteristics of Gtalk are similar to that of Games, making it difficult to classify Gtalk and Games. Because of this we added a new characteristic called Autocovariance. The autocovariance gives an array of values for the dependance or the similarity of the packet lengths in a sub-flow when lagged by a certain time t. The graphs shown below is the first covariance coefficient of Gtalk, Games and the Other traffic[11].

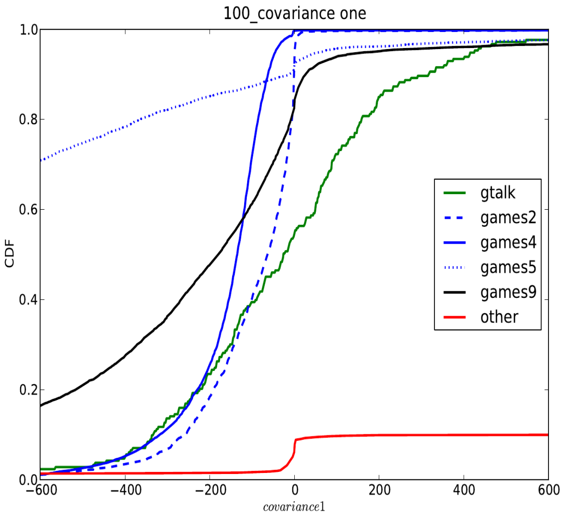

Figure 9: Autocovariance for a subflow size of 100.

The covariance graph shows a moderate statistical separation between the Games and the Gtalk traffic. Most of the Games traffic has an autocovariance that is less than 100. For Gtalk around 80% of the flows have an autocovariance coefficient less than 400. But the Other traffic has an autocovariance with a large variation. By using the first covariance coefficient and mean as attributes we were able to classify Gtalk and Games traffic.

The results obtained when training Gtalk and Other for the attributes Mean and the first Covariance is shown below.

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.998	0.001	0.967	0.998	0.982	Gtalk
0.999	0.002	1	0.999	0.999	Other

Table 1: WEKA-Statistical results for Gtalk and Other

```
=== Confusion Matrix ===
    a       b   <-- classified as
  501       1 |     a = Gtalk
   17   14991 |     b = Other
```

Figure 10: Confusion Matrix - WEKA Training

The result show that Gtalk can be classified against Other traffic with 0.97% precision and 0.98% recall.

The results obtained by testing Gtalk with Games using classifier trained against Gtalk and other traffic is shown below.

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.998	0.144	0.851	0.998	0.918	Gtalk
0.856	0.002	0.998	0.856	0.922	Other

Table 2: WEKA - Statistical results for Gtalk and Games

```
=== Confusion Matrix ===
    a      b   <-- classified as
  501      1 |     a = Gtalk
   88    524 |     b = Other(Games)
```

Figure 11: Confusion Matrix - WEKA Testing

The test results show that Gtalk can be classified with Games with a precision of 85% and a recall of 0.99%.

VII. Conclusion

Using generic attributes it is possible to identify Skype with 97% precision and recall of 86%. The attributes that were able to successfully identify Skype are the Mean, Index of Dispersion, Two Packet Difference and Inter-Arrival time. By changing only the attributes given to the classifier we can train a classifier with Skype version 3 and test the classifier with Skype version 4. Classification using sub-flows proved to be efficient in our experiment to identify real-time traffic. It should also be noted that sub-flow sizes should not be too small since there might be an insufficient number of packets within the sub-flow to identify the characteristics of traffic. Finally using different statistics for packet lengths as well as inter-arrival time the attributes identified enables us to identify Skype across version 3 and 4.

We also found that Gtalk and Games traffic could be classified with a precision of 85% and recall of 99%.

We believe these results show that generic classification of Skype and other traffic is possible and is a worthwhile area of future research.

Acknowledgments

Rozanna would like to thank Prof.. Grenville Armitage for the opportunity to work at CAIA. She would also like to thank Dr. Philip Branch and Dr. Jason But for their guidance and support.

References

[1] Philip Branch, Amiel Heyde, Grenville Armitage, 3-5 June 2009, Rapid Identification of Traffic Flows, *ACM NOSSDAV 2009*, Williamsburg, Virginia, USA.

[2] Philip Branch, Jason But, Tung Le, 11-14 October 2010,Rapid Identification of BitTorrent Traffic,35th Annual IEEE Conference on Local Computer Networks (LCN 2010), Denver, Colorado, USA,

[3] Skype website, http://forum.skype.com/index.php?showtopic=86573, accessed 22 July 2010

[4] SILK RFC, SILK Speech Codec draft-vos-silk-01, http://www.ietf.org/id/draft-vos-silk-01.txt, accessed 22 July 2010

[5] Salman Baset, Henning Schulzrinne, 2006 ,an analysis of skype peer-to-peer internet telephony protocol,IEEE Infocom.

[6] Thuy Nguyen, Grenville Armitage, 2006, Training on multiple subflows to optimise the use of Machine Learning classifiers in real-world IP networks in IEEE 31st Conference on Local Computer Networks, pp. 369-376. Tampa, Florida, USA.

[7] Ian Witten, Eibe Frank, 2005, Data Mining: Practical Machine Learning Tools and Techniques, Second Edition , page 363-368.

[8] http://www.ibm.com/developerworks/library/ac-adapt/index.html, accessed 29 July 2010

[9] University of Twente, Traffic measurement data repository, http:// traces.simpleweb.org, accessed February 2010.

[10] Jason But, Thuy Nguyen, Lawrence Stewart, Nigel Williams, Grenville Armitage, September 2007, Peformance Analysis of the ANGEL System for Automated Control of Game Traffic Prioritisation, 6th ACM SIGCOMM workshop on Network and system support for games (Netgames 2007), pp. 123-128. 19-20

[11] Autocovariance, http://w3eos.whoi.edu/12.747/notes/lect06/l06s02.html Accessed 05 July 2010