Classification of Application Traffic Using Tensorflow Machine Learning

Jee-Tae Park, Kyu-Seok Shim, Sung-Ho Lee and Myung-Sup Kim
Dept. of Computer and Information Science, Korea University
Sejong, Korea
{pjj5846, kusuk007, gaek5, tmskim}@korea.ac.kr

Abstract— **Applications are becoming more complicated and diverse as the network environment grows day by day. So, it is important to classify application traffic accurately. Although there are many ways to classify applications traffic, machine learning based approaches are becoming more efficient in nowadays. This is because machine learning methods are more appropriate than existing methods for accurate and efficient applications traffic classification. Payload signature methods have limitations to deal with various patterns and increasing application traffic complexity. In this paper, we propose a method for extracting flow features and a system for classifying applications traffic based on Machine Learning**

Keywords — Traffic Classification, Machine Learning, Learning Feature, Flow Feature Extraction, Normalization

I. Introduction

With the rapid Internet volume growth, network environment is daily becoming complex and diverse. So that application traffic patterns are becoming even more diverse and complex. In this situation, network traffic monitoring and analysis is essential for effective network operation and stable service provision. In order to analyze the network traffic, application traffic classification method must be preceded [6]. And, it is important to find appropriate methods for precise classification of application traffic [1,2,3,5]. The most widely known method on these days is the payload signature based classification method. We extract application traffic payload signatures and match the payload signature with other traffic patterns. The application traffic classification based on payload signature has many advantages with high accuracy and performance. However, there are also many problems of this method. It is computationally expensive for real time handling of large amounts of traffic on high speed network [4]. And this classification method has a limitation that it is difficult to cope with new patterns other than the existing ones [1,2,5].

The application traffic classification method based on the machine learning can solve these problems. Even if a new pattern of application traffic is found, it will be classified itself based on the patterns learned before [7]. Also many companies provide machine learning OpenAI tools so that people can use it freely. Among the machine learning tools, we use Tensorflow for this system. Because it is not only mostly well-known and also easy to use.

However, there are several cautions to classify application traffic based on machine learning [1]. First, it is important to apply appropriate learning features to be used in Machine Learning [7,8]. Inappropriate learning features result in poor accuracy compared to existing traffic classification methods. Second, the experiment should be conducted many times under various learning features. It is important to find the optimal learning features. Because the result greatly depend on the learning features. In order to find optimal learning features, many experiments should be conducted with various learning features.

The rest of the paper is organized as follows. In section II we discuss related work and the challenges discovered. In section III, we propose traffic classification system. We explain it briefly and divide this system into 2 part. One is learning feature extraction system, and the other is traffic classification system based on machine learning. In section IV, we evaluate this classification system through several experiments. Finally, we conclude these experiments and refer the future work in section V.

II. Related Work

There have been various studies on application traffic classification. The most common classification method among the various methods is a payload signature based classification method. This method extracts a unique signature for each payload of each application traffic and classifies it based on the extracted signature. Such classification based on payload signature is difficult to apply if the data part is encrypted [3]. A system that automatically generates signatures has been developed, which allows signatures to be generated even if the data is encrypted. Classification of application traffic with generated signatures gives high accuracy. However it gradually becomes difficult to flexibly cope with the complicated and

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2015R1D1A3A01018057) and by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. 2017-0-00513)

978-1-5386-1101-2/17/$31.00 ©2017 IEEE

diverse application traffic pattern [2,3].

Machine learning method can be solution of this problem. The greatest advantage of machine learning can cope with diverse situation even if a new situation occurs abruptly. Therefore, if machine learning is applied to application traffic classification, it can be flexibly coped with a complicated and diverse application traffic pattern [1].

Both methods result high accuracy and performance. But, payload signature method is not only expensive but also hard to cope with diverse traffic pattern. Machine learning method can solve these problems efficiently.

III. Proposed Classification System

In this paper, we propose an application traffic classification system based on machine learning. Figure 1 shows the overall structure of the proposed system. Proposed structures can be divided into 4 types.

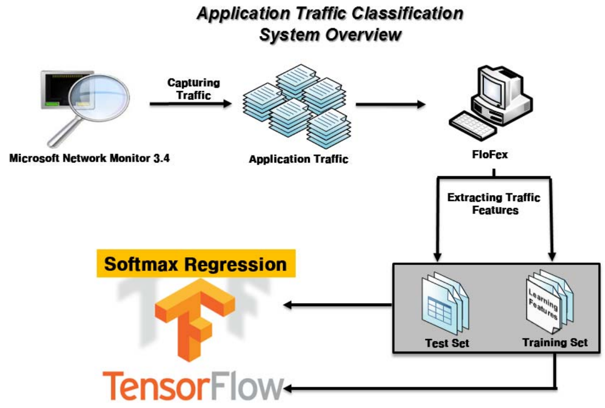

Figure 1. Proposed Traffic Classification System Overview

In the first step, collect application traffic. In this step, we capture from various applications through Microsoft Network Monitor 3.4. Application traffic should be captured in appropriate amount. If the amount of traffic is not adequate, it will not only give accurate classification results, but will also take a long time.

In the second step, extract the learning features of the collected application traffic. In this step, instead of the existing learning feature extraction system, the proposed learning feature extraction system is used. Few of learning features are extracted by using existing system. However proposed learning feature extraction system can solve this problem. It will be mentioned more detail in III-1. Extracted learning features of each application are divided into Train Set and Test Set. Train Set is a set of extracted learning feature to be learned through machine learning. Test Set is a set of extracted learning feature to be tested whether the classified traffic is correctly classified at the next stage.

In the third step, classified Train Set and Test Set are normalized. There are a number of ways to normalize. However, the results greatly depends on which normalization method is used. Therefore, it is also important to use sophisticated and appropriate normalization method in this step.

Finally, conduct several experiments with normalized Train Set and Test Set. We will use Softmax regression among the various classification methods of Tensorflow. We will refer each of steps in more detail in III-2.

III-1. FloFex - Learning Feature Extraction System

As mentioned earlier, it is very important to extract appropriate learning features. Originally extracting methods not only complicate but few learning features are extracted. Therefore, we propose a system that can extract learning features of traffic more efficient and various than other extracting methods. A system consists of 2 parts.

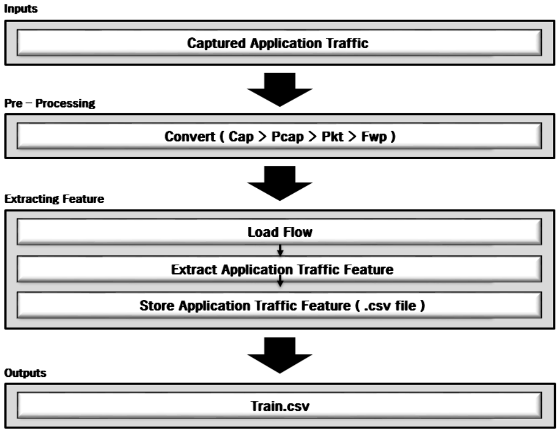
Inputs
Captured Application Traffic
Pre – Processing
Convert (Cap > Pcap > Pkt > Fwp)
Extracting Feature
Load Flow
Extract Application Traffic Feature
Store Application Traffic Feature (.csv file)
Outputs
Train.csv

Figure 2. FloFex System Structure

The structure of the FloFex is shown in Figure 2. First, transform the captured application traffic coming into the input into the proper form. Second, the learning feature of the application traffic is extracted based on the modified fwp file. Finally, the learning features of the extracted application traffic are stored in the output

The structure of the extracted learning features are shown in Figure 3. There are about 300 learning features that can be extracted from FloFex. They can be divided into main feature and sub feature. The reason for dividing into two parts is to explain and classify efficiently.

Main feature occupies most of learning features extracted by FloFex. In Main feature, 288 Features are derived. The main feature can be divided into 6 types according to time area (During 1 ~ 5 seconds, and Total time). Main feature also can be divided into 2 parts as packet size, and inter arrival time. Packet size corresponds a size of packets, and inter arrival time corresponds the time between packets in each flow. Each of them divided into total, forward, and backward, and each has 8 statistical values (max, min, median, sum, mean, variance, 1st_variance, 3rd_variance).

Sub feature can be divided into four major parts as shown in Figure 3 of Sub feature. In sub feature, 15 features are derived. First, packet count is the number of packets in each flow, which can be divided into total, forward, and backward. Packet count with payload is also the number of packets. But packet count with payload is the number of packets excluding the packets which payload length is zero. Second, the flow size, duration, and protocol correspond to the size (number of bytes), duration, and protocol of each flow. Source port, source address, destination port and destination address also correspond to the port number of source and destination and address of source and destination corresponding to each flow.

Finally, PPS (Packet Per Second) is defined as the number of packets sent per second. PPS is divided into PPS and FPPS BPPS according to total packet, forward, packet and backward packet as well as packet count. That is, they represent the percentage of total, forward, and backward packets sent for one second each.

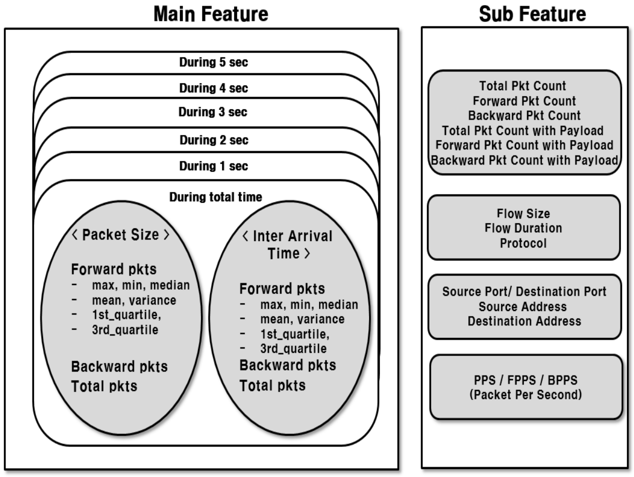

Figure 3. Learning Feature List Structure Extracted from FloFex

To summarize, there are 24 learning features can be extracted in each time zone (During 1~5 seconds, Total time). So, packet size and inter arrival time each has 144 learning features, resulting in total of 288 learning features. In the sub feature, there are 15 learning features. Therefore, total learning feature counts come out to 303.

III-2. Traffic Classification based on Machine Learning

As shown in Figure 1, there are 4 parts of application traffic classification system based on machine learning which is proposed in this paper. We collected application traffic and extract learning features by using FloFex. Then extracted learning feature is divided into Train Set and Test Set. Because each of 2 Sets has a different of role. Train Set is a Set for learning. It occupies for 70 ~ 80% of the extracted learning features. Test Set is a Set to verify learning. Test Set occupies the rest of the extracted learning features. Figure 4 is a simplified representation of the first and second step described in Figure 1.

In the third step, we normalize learning features which are extracted from FloFex. Because the difference between each values of extracted learning features is too large to classify. If we do not normalize these values, it classified inaccurately. Therefore, it is important to use an appropriate normalization method. Although there are many methods of normalization, we use Min-Max Normalization. Because, it is more simple and efficient than other methods. In the final step, the experiment is performed with the normalized Train Set and Test Set.

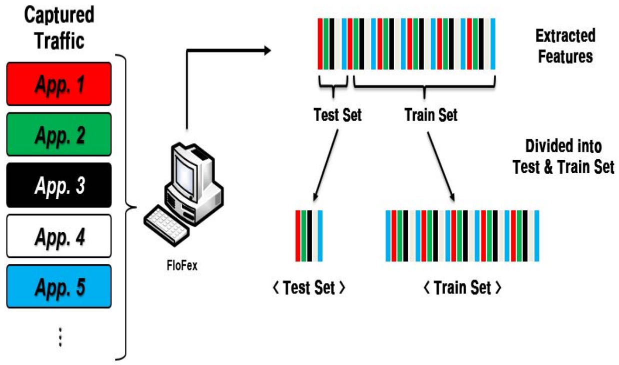

Figure 4. 1, 2 steps of Traffic Classification based on Machine Learning

First, train with the Train Set and test with the Test Set. There are many ways to learning and classifying by using Tensorflow. Among them, we will use softmax regression. Softmax regression is a classification method that is used mainly for three or more classification objects (Multinomial Classification). When there are 2 classification objects, we should use logistic regression (Binary Classification). It is more simple method than softmax regression. However, the number of application traffic which we should classify is more than 2. Therefore, it is suitable to use softmax regression.

Second, test with Train Set and check its accuracy. In this case, we should conduct many experiments to obtain the optimal learning rate and the number of loops with the highest accuracy. And apply them to the most optimal extracted learning features with the obtained learning rate and the number of loops. Then, we make an application traffic classification model based on machine learning.

IV. Evaluation

In order to prove the validity of the proposed system, we conduct many experiments. We collected the web browser application traffic. The web browsers used in this experiment are Chrome, Firefox, Internet Explorer, Swing, and Whale. All of these 5 web browsers application traffic are collected in same conditions.

The number of loops is the number of learning. Usually if the number of loops is bigger, the learning results is better. But it takes a lot of time to run many loops. Learning rate indicates the rate of learning. If the learning rate is too small, then it will spend a lot of time. On the other hand, if the learning rate is too big, then spending time will be shorten, but the results of accuracy will be poor. Therefore, it is important to find the optimal learning rate through experiments. Cost shows the difference between the actual value classified and the predicted value based on learning. Cost is the better when it converges to 0. Accuracy, on the other hand, represents the percentage of correctly classified traffic among the analyzed traffic. Accuracy is the better when it converges to 100%. The results of learning characteristics, learning rate and the number of loops used in this experiment are as follows.

Table 1. Traffic Classification Results using 5 Learning Features

L LR		2,000	20,000	200,000	10,000,000
0.01	C	1.545	1.511	1.503	1.478
	A	28.00%	30.89%	30.44%	31.25%
0.1	C	1.511	1.503	1.493	1.473
	A	30.89%	30.44%	29.78%	30.28%
1.0	C	1.503	1.493	1.479	1.481
	A	30.44%	29.78%	30.89%	30.89%
1.5	C	1.502	1.490	1.475	1.491
	A	30.44%	29.56%	31.56%	31.77%
3.0	C	1.499	1.486	1.489	1.477
	A	30.44%	30.44%	32.00%	32.03%
5.0	C	1.497	1.483	1.464	1.456
	A	29.56%	32.00%	32.67%	33.11%

L : The number of Loop / LR : Learning Rate
C : Cost / A : Accuracy

Table 1 shows the result of classification using 5 learning features. Learning features which we used in this experiment are flow size, flow duration, and packet count (total forward, backward). In this experiment, in most cases result poor accuracy. The result of this experiment shows that regardless of the number of loop, the accuracy is 30%.

Table 2. Traffic Classification Results using 14 Learning Features

L LR		2,000	20,000	200,000	10,000,000
0.01	C	1.446	1.139	1.013	0.864
	A	40.22%	43.78%	62.89%	67.13%
0.1	C	1.140	1.013	0.733	0.556
	A	43.78%	62.89%	80.44%	85.45%
1.0	C	1.013	0.732	0.395	0.249
	A	62.89%	80.44%	87.78%	94.22%
1.5	C	0.981	0.670	0.349	0.219
	A	65.11%	82.22%	89.33%	94.67%
3.0	C	0.907	0.563	0.262	0.186
	A	67.33%	85.33%	93.11%	95.56%
5.0	C	0.837	0.487	0.243	0.172
	A	72.22%	85.78%	94.22%	96.00%

L : The number of Loop / LR : Learning Rate
C : Cost / A : Accuracy

Table 2 shows the results of classification using 14 learning features. Learning features which we used in this experiment are flow size, flow duration, port number, protocol, and PPS, packet count, packet count with payload (total, forward, backward). When the number of loop is 2,000, the accuracy is almost 50%. However, when the number of loop is 10,000,000, the accuracy is 90%. Comparing with a previous experiment, its result improve especially in 10,000,000 loops. Thus, 14 learning features are more optimal learning features than the previous 5 learning features.

V. Conclusion and future work

In this paper, we proposed an application traffic classification system based on machine learning. We also designed a Flow Feature Extraction system (FloFex) for the proposed classification system. FloFex is able to extract more diverse and efficient than existing learning feature extraction system. Through various experiments, we confirm high accuracy and performance similar to existing traffic classification based on payload signature. And we also solve several problems of traffic classification based on payload signature.

In the future, we will try to find more optimal learning features than used in these experiments. We will also conduct various experiments with other application traffic to make precise traffic classification models.

References

[1] S.-H. Lee, K.-S. Shim, Y.-H. Goo and M.-S. Kim, “Application Traffic Classification using Tensorflow Machine Learning Tool”, Nov. 18. 2016, pp224-225

[2] J.-S. Park, S.-H. Yoon, H.-M. Ann, and M.-S. Kim "Classification of Internet Traffic Based on Payload Signature". May. 15-16, 2014, pp10-14

[3] Y.-H. Goo, K.-S. Shim, S.-K. Lee, M.-S. Kim "Payload Signature Structure for Accurate Application Traffic Classification" Proc of the Asia-Pacific Network Operations and Management Symposium (APNOMS) 2016, Kanazawa, Japan , Oct. 5-7, 2016

[4] S.-H. Lee, J.-S. Park, S.-H. Yoon and M.-S. Kim, “High performance payload signature based internet traffic classification system”, Proc. Of the Asia-Pacific Network Operations and Management Symposium(APNOMS)2015, Busan, Korea, Aug, 19-21, 2015.pp491-494

[5] J.-S. Park, S.-H. Yoon H,-M. Ann, M.-S. Kim," Internet Traffic Classification Based on Payload Signature". 2014 Communication Network Management Conference(KNOM2014), Chungnam National University, Daejun, May. 15-16, 2014, pp10-14

[6] Y.-H. Goo, S.-H. Lee, K.-S. Shim, M.-S. Kim "Traffic Classification Method Using Association Model of Network Flow", Journal of KICS, Vol.44 No.04, Apr. 2017, pp. 433-438

[7] Blum, Avrim L. and Pat Langley. "Selection of relevant features and examples in Machine Learning." Artificial intelligence 97.1 (1997): 245-271.

[8] Guyon, Isabelle, and André Elisseeff. "An introduction variable and feature selection." Journal of Machine Learning research 3.Mar (2003): 1157-1182.