A Novel Semi-Supervised Machine Learning Technique for Real-Time Network Traffic Classifications

Niyas N
M-Tech Scholar in Computer Science & Engineering
Calicut University
Calicut, India
niyasn.00009@gmail.com

Abstract— Traffic classification technique is an essential tool for network and system security in the complex environments such as cloud computing based environment. The state-of-the-art traffic classification methods aim to take the advantages of flow statistical features and machine learning techniques, however the classification performance is severely affected by limited supervised information and unknown applications. In unsupervised methods, different group of similar items called clusters are generated, but these clusters are need to be identified. For this we need some additional supervised information. In Classifier, a pre-labeled set of training instances are used to train the classifier. To make an accurate classifier this set of pre-labeled instances must be large, but it is impossible since new applications are emerging day by day. Also supervised method never detects unknown flows or intrusions. To tackle these problems, I designed a novel semi-supervised approach that integrates the advantages of both supervised and semi-supervised methods. This technique is applied over the real-time data to simulate the proper behavior of this new methodology.

Keywords-component; Packet classification; traffic classification; intrusion detection; k-means++; Naïve-Bayes Classifier.

I. INTRODUCTION

Traffic classification technique plays an important role in modern network security and management architectures. For instance, traffic classification is normally an essential component in the products for QoS control and intrusion detection; the application oriented network traffic classification can be effectively used to handle the problems of intrusions [1]. There exists a specific pattern for each type of network attacks such as denial of service attacks, worm propagation, intrusions, and spam spread; so to detect these attacks different pattern matching techniques can be applied [2]. In current scenario, Snort [3] and Bro [4] are present to monitor and manage networks, but these tools are failed to produce accurate results.

With the popularity of cloud computing, the amount of applications deployed on the Internet is quickly increasing and many applications adopt the encryption techniques. This situation makes it harder to classify traffic flows according to their generation applications. Traditional traffic classification techniques rely on checking the specific port numbers used by different applications, or inspecting the applications' signature strings in the payload of IP packets. These techniques encounter a number of problems in the modern network such as dynamic port numbers, data encryption and user privacy protection. The current research efforts have been focused on the applying machine learning techniques to the network traffic classification. For that different flow statistical features are derived from the packet. Machine learning techniques can efficiently use to find out structural patterns from applied data set and perform accurate traffic classification [5].

The flow statistical feature-based traffic classification can be achieved by using supervised classification algorithms or unsupervised classification (clustering) algorithms. In Clustering method, different clusters of similar items are generated. There are different clustering algorithms, among which only k-means produce much accurate clusters compared to others [6]. Also some predefined knowledge is needed to accurately map these clusters into known applications. This is a limitation since such a mapping produces many unmapped clusters for known apps. In supervised method, a pre-labeled set of instances for each application is needed to train the classifier. To make an accurate classifier this set of pre-labeled instances must be large, but it is impossible since new applications are emerging day by day [7]. Also supervised method never detects unknown flows or intrusions. Needs of accurate traffic classification that overcomes the limitation of both the supervised and unsupervised methods motivate me to propose this method.

In this paper, a new traffic classification framework is proposed that solves the drawbacks of both supervised and unsupervised methods. The major contributions for my work are summarized as follows:

- I propose a new a system model to incorporate flow correlation into a semi-supervised method, which

possesses the capability of unknown flow detection. This flow correlation is achieved using a pre-labeled set of instances.

- I proposed the compound classification to jointly identify the correlated flows in order to further boost the classification accuracy. This can be achieved with the aggregation of Naïve-Bayes predictions using majority vote rule.

The remainder of the paper is organized as follows: Section 2 reviews related work in traffic classification. A novel classification approach and the theoretical analysis are proposed in Section 3. Section 4 presents an experimental evaluation and the paper is concluded in Section 5.

II. RELATED WORK

Traditional way of network traffic classification can be done by analyzing transport layer port numbers and inspecting the payload bits. This approach is presently useful since majority of cloud based applications are peer to peer type and to secure information packets are usually encrypted. Now a day, most common approach is analyzing flow statistical features to classify network packet trace. For this different machine learning techniques such as supervised and unsupervised methods are used. J. Erman, M. Arlitt and A. Mahanti compares different unsupervised methods for traffic classification and found that both K-Means and DBSCAN work efficiently when compares to AutoClass and concluded that K-means provide better accuracy over other methods when k-value set to high [8].

In case of Supervised approach Moore and Zeuv indicates in their paper that with the simplest of Naive Bayes estimator they can able to achieve accuracy about 65% on specific flow classification and with some modifications this accuracy can be improved. The refinements are use of feature selection and feature discretization methods [9]. For feature selection a new method, Fast-Correlation Based Filter is introduced, which specifically select de-correlated features which are essential for proper classification [10]. Importance of Feature Discretization is explained in [11].

Classifiers are combined to achieve the best possible classification performance, the idea is not to rely on a single decision making scheme. Instead, all the designs, or their subset, are used for decision making by combining their individual opinions to derive a consensus decision. The ultimate goal of designing pattern recognition systems is for the task at hand [12].

Many semi-supervised methods are developed by different researchers [13][14], but all of them lack accuracy and forms unmapped clusters.

III. SYSTEM ARCHITECTURE

This section presents a novel semi-supervised approach to deal with the correlated flows in an effective way, which can significantly improve the classification performance even with a small set of supervised training data. This approach integrates both supervised and unsupervised learning. Initially a training model using Naïve-Bayes Classifier is designed, for this a pre-labeled set of captured packets of different applications are used. Then the real-time packets are captured and perform clustering to get similar group of flows. After that these clusters are mapped to corresponding application using the trained model, for that I uses a threshold value **T**. If the cluster's ratio is less than that of **T**, it is considered as unknown flow..

3.1 Clustering Process

Fig. 1 illustrates the classification process of our proposed scheme, which is focused on flow-level traffic classification. In the preprocessing, the system captures IP packets crossing a target network and constructs traffic flows by checking the headers of IP packets.

These captured packets are fed to feature selection and feature discretization process to find out most appropriate flow statistical features. Then these packets are clustered using k-means++ algorithm based on selected features.

K-means ++ Algorithm

1. Initialize seed points as centroid of clusters manually
   //In ordinary k-means seeds are randomly assigned
2. Assign the selected instance to the cluster that has the closest centroid
3. Recalculate the positions of the centroids
4. If the positions of the centroids didn't change go to the step 5, else go to step 2.
5. End

. That is, in k-means++ if there are ***n*** applications, then ***n*** seed points are assigned corresponding to each application. It will form ***n*** clusters. If any intrusions or unknown flows are present, it will become part of some clusters. A pre-defined trained model is used to map these clusters to the specified applications.

3.2 App-Based Cluster Mapping

K-means++ algorithm produces the clusters corresponding to specified seeds. If any intrusions occurs or any new application come to exists (unknown flow), then the flow corresponding to them is also becomes part of the generated clusters.

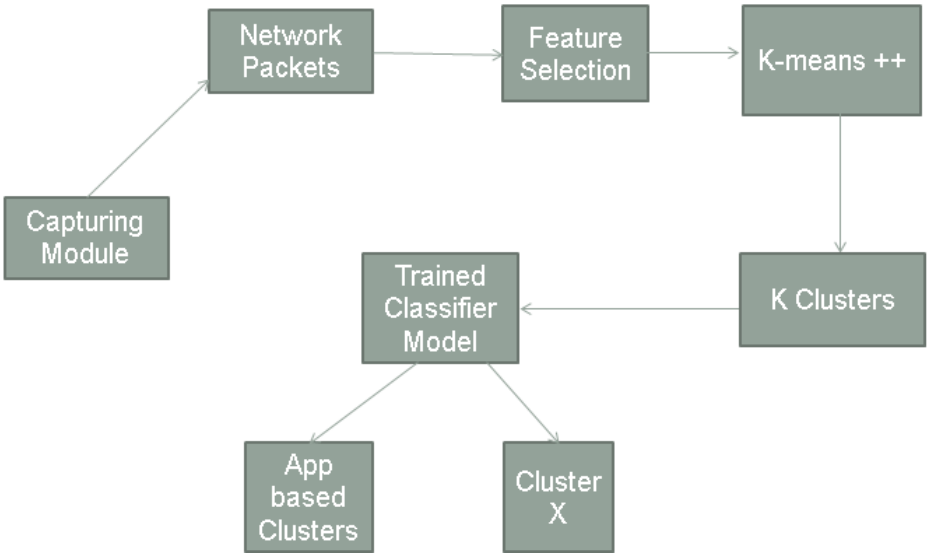

Figure 1: System Architecture

To get the accurate mapping of clusters, a Naïve-Bayes training model is used. Each cluster contains similar group of flows, and each flow is considered as instance of dataset. The training model finds out the probability of each instance by deriving its flow features and maps each instance to a specified class. Then apply majority vote rule [12] to aggregate the Naïve-Bayes predictions. If the ratio of majority vote with total number of instances in the specific cluster is less than a specific threshold **T**, then that cluster is marked as un-known Cluster B. Otherwise it is the cluster of application class determined through majority voting.

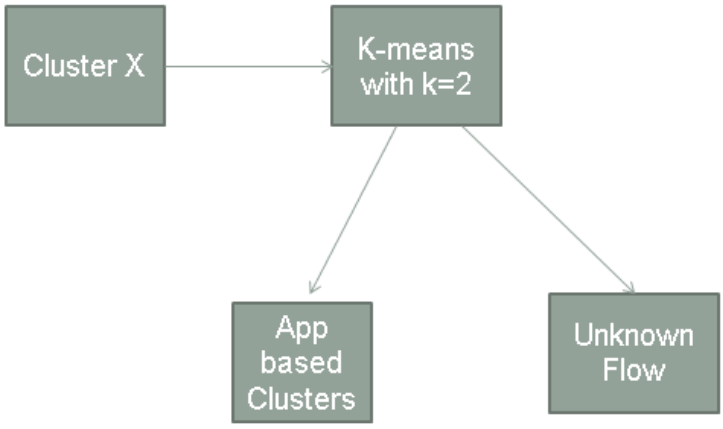

Figure 2: Unknown Flow Extraction

The cluster B consists of both the instances of specified application and that of unknown flows or intrusions. To extract the instances of application from that cluster, a second level clustering technique is implemented. The method is explained in figure 2.

After passing through k-means with k=2 algorithm, two clusters are produced and these clusters are again fed to training model to evaluate. Then the cluster which is less than the specified threshold **T** is considered as unknown flow.

IV. EXPERIMENTAL EVALUATION

In all previous research papers about the network traffic classifications performed the evaluations on pre-captured dump network trace. So that the accuracy they proposed in their model is not applicable in case of real scenario. To overcome this drawback the proposed model is evaluated using real time captured packets. The proposed method of classification is evaluated by considering three real time applications: a) video streaming application, b) chatting application and c) remote management application.

The classification performance can be measured using following metrics:

- Overall accuracy – It is measured as a ratio between sum of correctly classified instances with sum of testing instances.
- F-measure – It is used for evaluating performance of specific application class.

$$F - measure = \frac{2 \times precision \times recall}{precision + recall}$$

- Precision – It is measured as the ratio between correctly classified instances with all predicted instances.
- Recall – It is measured as the ratio between correctly classified instances with all instances.

Table 1: Unidirectional Features

Type	Features	Count
Packets	Number of packets transferred in unidirection	2
Bytes	Volume of bytes transferred in unidirection	2
Packet Size	Min., Max., Mean and Std Dev. of packet size in unidirection	8
Inter-Packet Time	Min., Max., Mean and Std Dev. of Inter Packet Time in unidirection	8
	Total	20

Initially a classifier model using Naïve-Bayes predictions is developed. For that certain flow features which are shown in Table 1 are derived. These features are also used in testing phase. I created the training model by taking 20 instances of each application. This set is very small, when compared to other methods for classification.

Then during testing phase, the real time packets are captured from the wired network and fed these traffic flows to K-means++ algorithm after deriving needed features. The three clusters are formed, and then these clusters are mapped in to corresponding applications by means of training model.

V. CONCLUSION

In this paper, I am proposing a new semi-supervised traffic classification scheme which can effectively improve the classification performance in the real-time situation and also in the situation that only few training data are available. The proposed scheme is uses a two level clustering mechanism integrated with Naïve-Bayes classifier to detect intrusions or unknown flows. This approach can be effectively used to classify packets in a cloud-based environment, so that network management becomes easier. Here a new K-means algorithm called K-means++ algorithm is implemented, which overcomes the drawback of ordinary K-means. Ordinary K-means never reach local optima, so accurate clusters not formed. But in our proposed architecture using K-means++ with NB-Classifier model produces more accurate classification.

REFERENCES

[1] T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," Commun. Surveys Tuts.,vol. 10, no. 4, pp. 56–76, 4th Quarter 2008.

[2] Y. Xiang, W. Zhou, and M. Guo, "Flexible deterministic packet marking: An ip trace back system to find the real source of attacks," IEEE Transactions of Parallel Distributed System., vol. 20, no. 4, pp. 567–580, Apr.2009.

[3] Snort 2011 [Online]. Available: http://www.snort.org/

[4] Bro 2011 [Online]. Available: http://bro-ids.org/index.html

[5] N. Williams, S. Zander, and G. Armitage, "A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification," Proc ACM SIGCOMM, vol. 36, pp. 5-16, Oct. 2006

[6] J. Erman, A. Mahanti, and M. Arlitt, "Internet traffic identification using machine learning," inProc. 2006 IEEE Global Telecommunications Conference, pp. 1–6..

[7] A. Finamore, M. Mellia, and M. Meo, "Mining unclassified traffic using automatic clustering techniques," in Proc. 2011 TMA International Workshop on Traffic Monitoring and Analysis, pp. 150–163.

[8] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms," in Proc. SIGCOMM Workshop on Mining Network Data, New York, 2006, pp. 281–286

[9] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in SIGMETRICS Perform. Eval. Rev., Jun. 2005, vol. 33, pp. 50–60

[10] Lei Yu, Huan Liu, “Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution”, Department of Computer Science & Engineering, Arizona State University, Tempe, AZ 85287-5406, USA
[11] Y.-S. Lim, H.-C. Kim, J. Jeong, C.-K. Kim, T. T. Kwon, and Y. Choi, “Internet trafficclassification demystified: On the sources of the discriminative power,” inProc. 6th Int. Conf., Ser. Co-NEXT’10, New York, 2010, pp. 9:1–9:12, ACM.
[12] J. Kittler, M. Hatef, R. Duin, and J. Matas, “On combining classifiers,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 3, pp. 226–239, Mar. 1998.
[13] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, “Offline/real time traffic classification using semi-supervised learning,” Performance Evaluation, vol. 64, no. 9-12, 4 pp. 1194–1213, Oct. 2007
[14] Y. Wang, Y. Xiang, J. Zhang, and S.-Z. Yu, “A novel semi-supervised approach for network traffic clustering,” in Proc. Int. Conf. Network and System Security, Milan, Italy, Sep. 2011, pp. 169–175