Unknown Network Detection using Machine Learning Method in Packet Sniffing

Annu Ailawadhi , Anju Bhandari
M.Tech student, N.C. College Israna Panipat
Associateprofessor at N.C. College Israna Panipat

Abstract: Packet sniffing is the increased concern in this cyber era. any hacker or intruder can monitor what data is going on in the network. This raises a concern to detect and avoid these intruders. These are in the form of a botnets or small softwares which keeps eye on network traffic. In this work we provided a solution to detect these intruders and monitor the traffic securely.NIMA and MAWI dataset is used for network analysis and machine learning classifiers like SVM, KNN and navie bays are applied and compared. A pre-processing of attributes selection is done before feeding the data into classifiers.

_____*****_____

I. Introduction

Network traffic classification is a crucial task for a spread of network-related areas, together with network management, police work, and security. Traffic classification has historically been performed by inspecting port numbers. However, typically|this can be} often

ineffective because of the quantity of applications victimization non-unique and non-standard port numbers [1]. Deep-packet scrutiny avoids reliance on port numbers, however demands Associate in Nursing up-to-date information of application signatures and has vital machine quality, typically creating the approach unworkable for real-world use [2]-[3].

Machine learning techniques are gaining quality for his or her ability to effectively classify network applications victimization solely applied math flow options [1]-[3] and while not the drawbacks of a lot of ancient approaches. The open drawback we have a tendency to address is the way to improve the accuracy of traffic classification from applications that are troublesome to classify victimization solely applied math traffic flow properties.

In this paper, we have a tendency to apply a supervised machine learning technique to mechanically determine network applications victimization solely applied math traffic flow properties. Our approach is predicated on a number one supervised traffic classification approach [4], which may handle flows generated by unknown applications. we have a tendency to propose 2 step to the current methodology so as to more increase its effectiveness. First, our approach introduces Associate in Nursing alternate rule for distinctive applications, Second, we have a tendency to propose introducing feature choice into the system model. supported Associate in Nursing empirical analysis on a regular benchmark dataset, we have a tendency to show that our approach has Associate in Nursing accuracy of ninety nine.9%, a rise of over 4WD against the technique on that it's primarily based [4]. to boot, our approach improves the classification performance on each class.

Current research into traffic classification has shown various supervised, unsupervised, and semi-supervised machine learning techniques to be viable approaches.

Supervised machine learning approaches [5], [6] have been shown to achieve particularly high classification effectiveness. However, these approaches can only predict predefined classes found in the training data. Unsupervised learning approaches [7]-[8] classify from clusters of unlabelled training flows. While using unlabelled datameans they can handle known and unknown classes, mapping clusters to classes remains a key challenge.Semi-supervised approaches aim to address the problems of both supervised andunsupervised approaches. Erman et al. [2] developed an effective semi-supervisedapproach for classifying network applications, combining K-Means clustering withprobabilistic assignment. Using a small set of labelled flows with a larger unlabelledset, clusters with labelled flows can automatically be mapped to classes. Clusterswithout labelled flows represent unknown classes. The key advantage of this technique is simple class mapping and handling of unknown classes. With few labelled instances, however, clusters are often incorrectly labelled "unknown". A recent extension to this approach by Zhang et al. [4] countered this weakness by automatically extending the labelled portion of training data. This was done by identifying correlated flows – flows sharing the same destination IP address and port, and protocol – and sharing labels between them. This approach was shown to significantly increase thelabels available and thus better label clusters. Furthermore, applying compound classification to correlated test flows further improved effectiveness. It was shown tooutperform standard and state-of-the-art machine learning algorithms, including decision trees, K nearest neighbours, Bayesian networks, and the Erman et al. approach.While the Zhang et al. approach is a leading semi-supervised approach for traffic classification, certain traffic classes still proved challenging to identify. We aim totarget these classes for an overall more consistently effective classifier

II. Proposed Work

This work proposes supervised machine learning algorithm for improved classification of known and unknown network traffic flows. In this work we have taken two datasets NIMS and MAWI datasets which are used for machine learning algorithm and for testing the machine learning model using both datasets using four main classifiers which are K Nearest Neighbour (KNN) Classifier, Support Vector Machine (SVM) classifier, Naïve Bayes classifier and RUSBoost classifier. The training model made by using NIMS datasets is used to identify unknown labels of MAWI dataset such that it performs cross dataset testing and labelling for unknown network traffic flows.

Overall work can be divided into following steps for better understanding.

We have divided our work in eight sub cases which are

1. Extracting database from web links and make dataset usable for learning algorithm.
2. Analysing dataset, its features and labels.
3. selecting the best features by using feature selection algorithm
4. Divide NIMS datasets in different combination which are more probable to exist in network traffic flows.
5. Dividing data into testing and training randomly with ratio 80:20for each combination.
6. Create training model using four classifiers.
7. Test the testing data using training model created by all four classifiers
8. Test the MAWI dataset for unknown labels using trained model.
9. Comparison of output test labels from real labels and discussion.

The dataset has 22 attributes, all of which may not contribute to improve the accuracy, so feature selection is a pre-processing step which iteratively selects the features which contribute for better accuracy. We used sequential forward feature selection method which is default in MATLAB machine learning toolbox. It selects every feature and check the accuracy for classifier defined in its objective function. Then step forward to check next feature and ends when the improvement in accuracy is stopped. These 22 features and notations are given in table2.1.

Table 2.1 Network traffic flow features

S.No.	Feature Name	S.No.	Feature Name
1	Min_fpktl	13	Min_biat
2	Mean_fpktl	14	Mean_ biat
3	Max_fpktl	15	Max_ biat
4	Std_fpktl	16	Std_ biat
5	Min_bpktl	17	Duration
6	Mean_bpktl	18	proto
7	Max_bpktl	19	Total_fpackets
8	Std_bpktl	20	Total_fvolume
9	Min_fiat	21	Total_bpackets
10	Mean_ fiat	22	Total_bvolume
11	Max_ fiat		
12	Std_ fiat		

Their notations are

Fpktl-forward packet length	Proto-protocol
Bpktl-backward packet length	Total_bpackets-Total backward packets
Fiat-forward inter-arrival time	Total_bvolume-Totalbackward packet volume
Biat-backward inter-arrival time	Total_fvolume- Total forward packet volume
Total_fpackets-Total Forward packets	

by using the final selected features, the whole data is divided into training and testing set and classified accordingly. The trained model with best classification accuracy is saved and used for real traffic classification.

We have used four classifers which are K Nearest Neighbour (KNN) Classifier, Support Vector Machine (SVM) classifier, Naïve Bayes classifier and RUSBoost classifier.

The K nearest neighbours algorithm is a non-parametric method which is used for classification and regression. In both cases, input must consists of the k closest training examples in feature table. For KNN classification, the output is a class membership. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its K nearest neighbours. If K=1, then the object is simply assigned to the class of that single nearest

 ISSN: 2321-8169
43 – 48

neighbour. The KNN algorithm is among the simplest of all machine learning algorithms.

In machine learning , SVM (Support Vector Machines) are used as supervised learning models with associated learning algorithms that analyse dataset which is used for classification. We have a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm build a model which is representation of the examples as points in space properly mapped so that the examples of the separate categories are divide by a clear gap that is as wide as possible. New examples are then mapped into that very same space and predicted examples belong to a category based on which side of the gap they fall.

Other two classifiers are Naïve Bayes and RUSBoost classifiers which are simple probabilistic classifier. Naïve Bayes classifiers are a family of simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Once traffic data feature table is formed it is divided randomly into training and tetsing data in the ratio of 80:20. All four multi class classifier is used to create a training model using training data and testing data is used to test the label using that trained model.

Output labels of testing data is compared with predicted labels from trained model created using our proposed method with all features of image and calculates all four performance evaluation parameters. A flow chart for the proposed work is shown in figure 2.1.

III. Results

NIMS and MAWI datasets have very large dimension (NIMS-713851x23), (MAWI-500000x23), which means it has 22 features and last column is label. Four classifier are used to create trained model using NIMS training data. As we can see NIMS datatset is very large which takes very much to process , so we have randomly chosendata for each class in almost same ratio and then select randomly training and testing data from it. Similarly for MAWI dataset , as it is very large so we have takenrandomlyreduced no of samples to test from classifier trainedmodel.

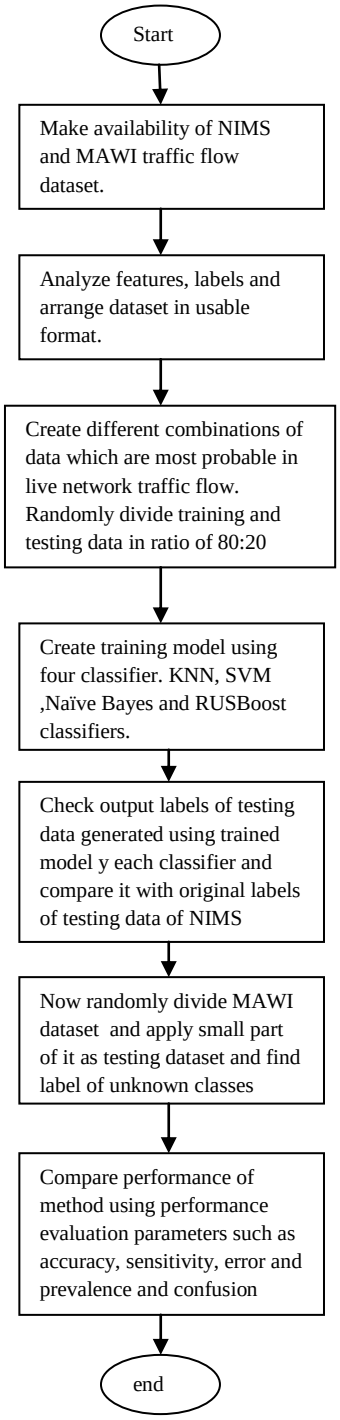


Figure: 2.1: Flow chart of proposed Method

The data is pre-processed before using it finally and all labels are converted into numeric for 1-11 classes as classifiers understand only quantitative data. Three different cases have been considered following the same methodology as discussed in previous chapter. The classifier is first trained with NIMS dataset and tested on MAWI dataset. Since both have same number of attributes and classes.

 ISSN: 2321-8169
43 – 48

Case1: Only two classes are considered- unknown network flow or not

These two classes are labelled as SSH and NOTSSH qualitatively and quantitatively these are 0 and 1. We have created training model using four classifiers (SVM, KNN, Naive Bays and RusBoost) for these binary class datasets. Their accuracy comparison is shown in figure 3.1 and a table is shown in table 3.1 for this binary classification.

Table 3.1: Binary Classification performance for Unknown network flows

Performance Measure	KNN Classifier	SVM Classifier	Naïve Bayes	RUSBoost Classifier
Accuracy	.9993	1.0000	0.9233	0.5838
Sensitivity	0.9990	1.0000	0.9950	1.0000
Error	0.0007	0	0.0767	0.4162

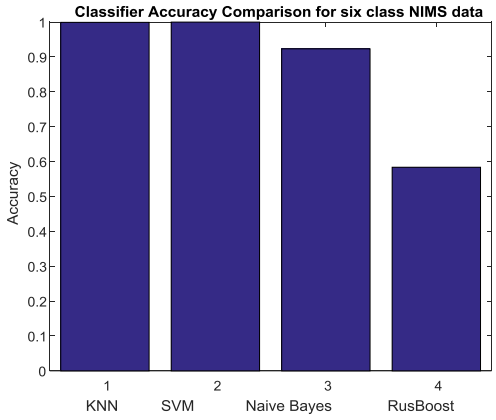
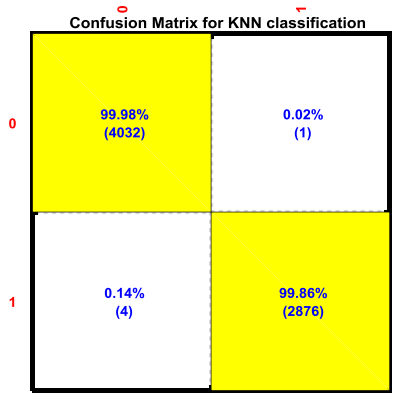


Figure 3.1: Accuracy comparison for binary classifier

By analysing the results it can be said that the SVM classifier is performing very well in terms of detection the malicious network flow. A confusion map for SVM and KNN will support our point. We are not showing here the other two classifier's confusion map as their performance very low than these.



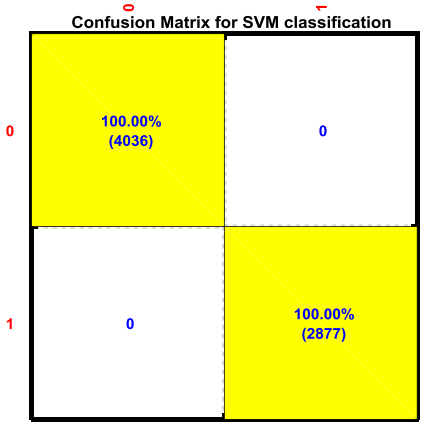


Figure 3.2 Confusion matrix for KNN and SVM Classifiers

The value in yellow box shows the % of correct detection and in the corresponding row, it is % of false detection. Confusion map shows the detected samples number and percentage for both classes individually.

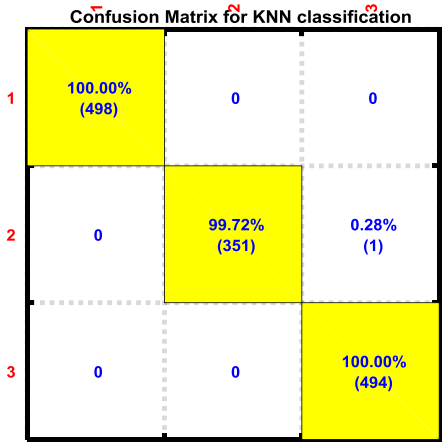
CaseII: Three classes are used for network flow detection

Here DNS, FTP, Shell(SSH) classes are taken with numeric labels 1,2,3. The performance table is shown in table 3.2.

Table 3.2: Three classes Classification performance for Unknown network flows

Performance Measure	KNN Classifier	SVM Classifier	Naïve Bayes	RUSBoost Classifier
Accuracy	0.9993	0.9993	0.9985	0.3705
Sensitivity	1.0000	1.0000	1.0000	1.0000
Error	0.0007	0.0007	0.0015	0.6295

Since the accuracy is equal for KNN and SVM classifier and a very good percentage is achieved, So confusion map are shown in figure 3.3.



 ISSN: 2321-8169
43 – 48

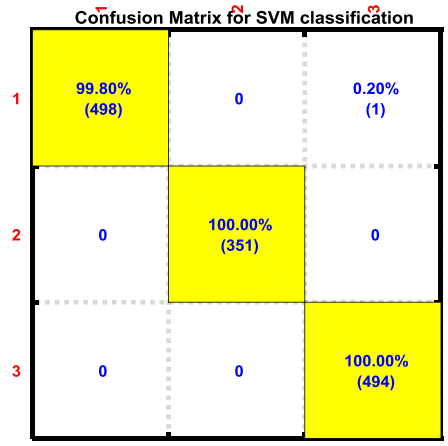


Figure 3.3 Confusion matrix for KNN and SVM Classifiers

Case III: Here all 11 classes , 6 for SSH and 5 for NOTSSH classes are taken as numeric 1-11.

The confusion matrix plot is shown as

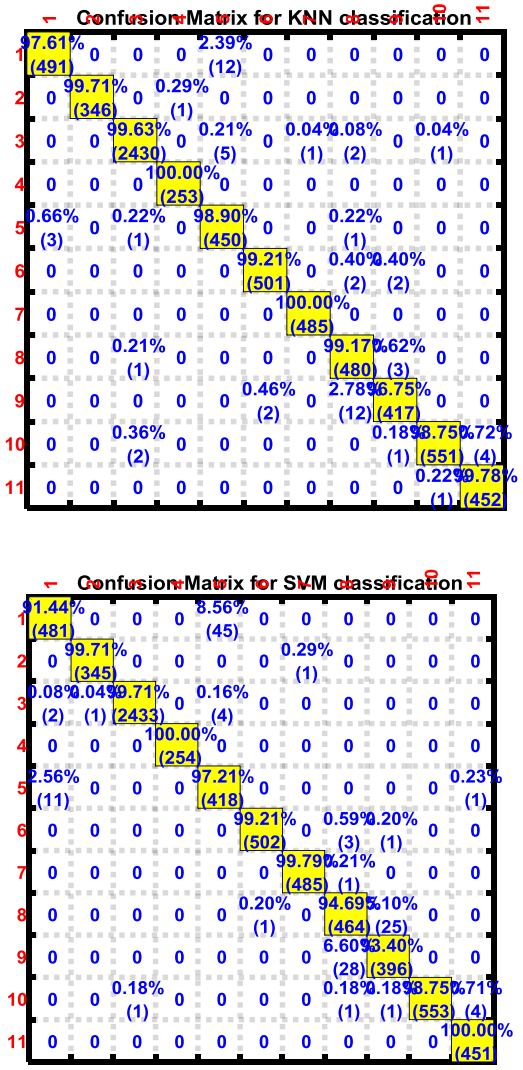


Figure 3.4 Confusion matrix for KNN and SVM Classifiers

On analysing the confusion map, KNN classifier is performing better than the SVM classifier for individual number of classes, though the difference is trivial. the overall performance of all classes is given in table 3.3.

Table 3.3: All classes Classification performance for Unknown network flows

Performance Measure	KNN Classifier	SVM Classifier	Naïve Bayes	RUSBoost Classifier
Accuracy	0.9918	0.9811	0.9682	0.3521
Sensitivity	0.9939	0.9737	0.9960	0
Error	.0082	0.0189	0.0318	0.6479

The RusBoost classifier's performance is least in every case in this case. SVM turned out to be best classifier on the ground that in case 3 the differences are very trivial from KNN.

IV. CONCLUSION

In this work a comprehensive study of packet sniffingdata analysis and classification has been carried out. In order toperform the above-mentioned investigation, we have used NIMS dataset from (http://www.cs.dal.ca/~riyad/DataSets/NIMS/NIMS.arff.zip), to create training model and tested our model on NIMS dataset and MAWI dataset is downloaded from (http://www.cs.dal.ca/~riyad/DataSets/MAWI/MAWI.zip) . For the sake of accuracy, we have used four classifiers which are K Nearest Neighbour (KNN) Classifier, Support Vector Machine (SVM) classifier, Naïve Bayes classifier and RUSBoost classifier. We have first created a training model using all four classifiers and firstly tested the system using NIMS dataset and then using MAWI dataset, where cross checking of dataset is performed. Previously network data traffic classification is done by semi-supervised approach , but here we have used supervised machine learning approach for improved classification of known and unknown network traffic flows. After testing the results using our approach it is concluded that SVM classifier works best in all four classifiers and predicted labels by SVM are having least error and highest accuracy. KNN classifier also works good under certain circumstances when testing dimension is small and classes of label is less. Network traffic flows are labelled as SSH and NOTSSH classes further for NIMS dataset SSH have 6 labels which are local tunnelling, remote tunnelling, SCP, SFTP, Shell and X11 and NOTSSH traffic is further divided in to 5 labels which are DNS, FTP, HTTP, TELNET, Lime (P2P).

References

[1] Karagiannis, T., Broido, A., Faloutsos, M.: Transport layer identification of P2P traffic. InACM SIGCOMM Conference on Internet Measurement. pp. 121 -134 (2004).

[2] Erman, J., et al.: Offline/realtime traffic classification using semi-supervised learning. Performance Evaluation. Vol. 64(9), pp. 1194-1213 (2007).

[3] Williams, N., Zander, S., Armitage, G.: Evaluating machine learning algorithms for automated network application identification. Center for Advanced Internet Architectures(CAIA), Technical Report B, 60410. (2006).

[4] Zhang, J., Chen, C., Xiang, Y., Zhou, W., Vasilakos, A. V.: An effective network trafficclassification method with unknown flow detection. IEEE Transactions on Network andService Management,. Vol. 10(2), pp. 133-147 (2013).

[5] Erman, J., et al.: Offline/realtime traffic classification using semi-supervised learning. Performance Evaluation. Vol. 64(9), pp. 1194-1213 (2005).

[6] Auld, T., Moore, A. W., Gull, S. F.: Bayesian neural networks for internet traffic classification. IEEE Transactions on Neural Networks. Vol. 18(1), pp. 223-239 (2007).

[7] McGregor, A., Hall, M., Lorier, P., Brunskill, J.: Flow clustering using machine learningtechniques. In PAM. pp. 205-214 (2004).

[8] Erman, J., Arlitt, M., Mahanti, A.: Traffic classification using clustering algorithms. InSIGCOMM Workshop on Mining Network Data. pp. 281 -286 (2006).

[9] Nguyen, T. T., Armitage, G.: A survey of techniques for internet traffic classification usingmachine learning. IEEE Comm. Surveys and Tutorials, Vol. 10(4), pp. 56-76 (2008).

[10] Williams, N., Zander, S., Armitage, G.: A preliminary performance comparison of fivemachine learning algorithms for practical IP traffic flow classification. ACM SIGCOMMComputer Communication Review. Vol. 36(5), pp. 5-16 (2006)

[11] http://www.cs.dal.ca/~riyad/DataSets/NIMS/NIMS.arff.zip

[12] http://www.cs.dal.ca/~riyad/DataSets/MAWI/MAWI.zip