International Journal of
Intelligent Systems and
Applications in Engineering
ISSN:2147-6799 www.atscience.org/IJISAE
Advanced Technology and Science
Original Research Paper

Network Traffic Classification via Kernel Based Extreme Learning Machine

Fatih Ertam*[1], Engin Avcı[2]

Accepted 3rd September 2016

Abstract: The classification of data on the internet in order to make internet use more efficient has an important place especially for network administrators managing corporate networks. Studies for the classification of internet traffic have increased recently. By these studies, it is aimed to increase the quality of service on the network, use the network efficiently, create the service packages and offer them to the users. The first classification method used for the classification of the internet traffic was the classification for the use of port numbers. This classification method has already lost its validity although it was an effective and quick method of classification for the first usage times of the internet. Another classification method used for the classification of network traffic is called as load-based classification or deep packet analysis. This approach is based on the principle of classification by identifying signatures on packets flowing on the network. Another method of classification of the internet traffic which is commonly used in our day and has been also selected for this study is the kernel based on extreme learning machine based approaches. In this study, over 95% was achieved accuracies using different activation functions.

Keywords: *Machine Learning, Classification, Extreme Learning Machine, Network Classification, Kernel Activation Function.*

1. Introduction

Traffic classification methods are used to provide the efficient realization of the data traffic on network resources, to do user analysis by using network data, to manage and plan network resources, to detect the attacks and the abnormalities on the network [1, 2]. Recently, the network traffic classification has been frequently used in order to improve service quality in big networks [3, 4], use the network effectively, develop new service packets and perform internet traffic analysis [5]. Internet traffic analysis can be done both on-line and offline. In the on-line traffic analysis, each data packet on the network are captured and analysed [6, 7]. In the offline traffic analysis, network traffic flow is firstly captured and stored; then the stored flow is analysed and classified [8]. In this paper, offline traffic analysis was performed. In the literature, three kind of classification technique, including port, payload and machine learning based, have been used.

Port based classification is performed by comparing the port information retrieved from flow data with the port numbers of protocols determined by Internet Assigned Numbers Authority (IANA) [9]. For instance; port 80 is used for http, and port 23 for telnet traffic. Especially with the widespread use of point to point (P2P) applications, this method has started to lose his functionality as some applications use non-standard port numbers to escape from firewall and network security tools and some use port hiding and dynamic port methods [10-13]. Payload based classification is based on the principle of Internet traffic classification by analysing TCP/UDP packet loads. The analysis of loads is performed by determining whether the known applications contain characteristic signatures [10, 14].

When the packets are not encrypted, it works quite successfully. However, because of the following reasons, this classification technique is not much preferred today:

- It causes privacy and security concerns,
- Some applications communicate by using encrypted packets,
- It can only make assessment based on the signatures which are experienced by the previous classification methods,
- As it requires high processing and storage capacity, it is not suitable for real-time classification [11-13].

The network classification method via machine learning algorithms is the most popular traffic classification method at the present time. In the studies, the classification is usually performed using supervised and unsupervised learning algorithms. Supervised learning algorithms perform classification by using the classification analysis methods in data mining; and unsupervised learning algorithms perform classification by using the clustering analysis methods. Machine learning algorithms perform the process of network traffic classification in two steps. In the first step, it forms a classification model; and in the second step, it performs the classification. Statistical methods and calculations are usually utilized when performing the classification process.

Machine learning based classification method uses the following TCP and UDP statistical attributes of the flow during the flow-based classification:

- Total size statistics,
- Total number of forward and backward packets,
- Total amount of forward and backward byte,
- The transit time between packets,
- And flow time.

There are many studies on the field of Internet traffic classification by using machine learning. In their study, L. Yingqiu et al. tried to classify network traffic on the original and log-transformed data

[1] *Firat University, Turkey*
** Corresponding Author: Email: ertam.f@gmail.com*
Note: This paper has been presented at the 3rd International Conference on Advanced Technology & Sciences (ICAT'16) held in Konya (Turkey), September 01-03, 2016.

set by using K-means algorithm [13]. In J. Erman's study, clustering algorithms of Autoclass, K-means and DBSCAN, were used [15]. S. Zander performed network classifications by applying Exception Maximization (EM) algorithm and method of attribute selection on WAND Research Group's open data sets and different data sets [16]. S. Zander et al. and S. Agrawal et al. made comprehensive comparisons about classification algorithms by using algorithms and attribute selection methods like C4.5, Bayes Net, Naïve Bayes [16-20]. Apart from port, payload, and machine learning based classifications in the literature, T. Karagiannis et al. developed an important classification method by using host service providers instead of TCP and UDP protocols [6]. L. Bernaille et al.'s classification technique which was performed by using unsupervised learning method and checking only the first few TCP packets is among the important studies in the literature [22]. Recently, ELM pattern recognition, which was suggested by Huang et al., has aroused great interest in the fields of machine learning and data mining; and a lot of applications have been performed regarding the issue [21,23-25,35]. ELM was suggested as a newly learning algorithm for Single-hidden Layer Feedforward Neural Networks (SLFNs) [23, 27]. During the learning process, SLFNs refreshes network loads based on the gradient. However, in ELM, input sizes and biases are random selected, and output sizes are calculated with an analytical method contrary to SLFNs. In this case, ELM gains the advantages of a fast learning process, a good generalization performance and a low computational load [27-29]. A higher accuracy percentage was obtained with Tangent sigmoid and triangular basis among activation functions which were used for the classic ELM algorithms in this paper. With kernel based extreme learning machine algorithm, which used Radial basis and polynomial activation functions rather than classic ELM, the accuracy percentage was observed to be even higher. In this paper, Internet traffic classification process has been shown using KELM faster and high accuracy.

In the second part of this paper, working principles of ELM algorithms were mentioned. The third part of the paper mentioned how the data was obtained and which data were used for the classification in experimental studies. In addition, different classification algorithms were compared.

2. Proposed Methodology

Moore et al [34] used the data received from the Cambridge University campus in this study. The most important factor in this selection is to enable to make comparison of studies previously carried out by using these data with the methods used in this study. One of the important factors is the use of data having flows belonging to different classes. In addition, the use of existing data that everyone can reach will provide a basis to get more reliable results.

2.1. Extreme Learning Machine

Extreme Learning Machine was recommended in G.B. Huang et al. [28-31]. ELM used from 2004 onwards for training the Single-hidden Layer Feedforward Neural Networks (SLFNs) [23, 24]. Dissimilar from the extensive understanding of training SLFNs, ELM employs randomize computational nodes in the hidden layer and computes its output weights analytically by solving a general linear system equation. Later, ELM theory was extended to the "generalized" SLFNs, where the hidden nodes need not be neuron alike. The generic architecture SLFNs shows in (Figure.1)

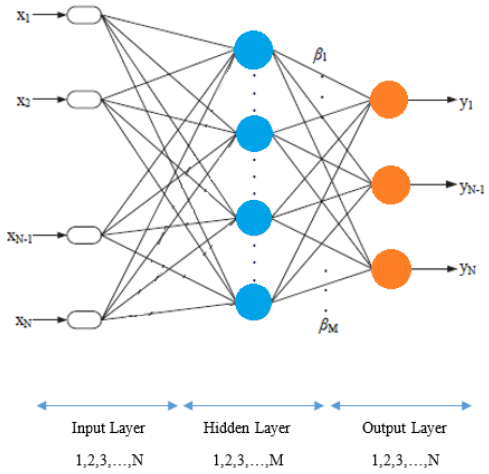

Figure. 1. Single-Hidden layer feed forward network architecture

Given N arbitrary training examples $\{(x_j, t_j)\}_{j=1}^{N} \subset R^d XR^m$, the output of the generalized SLFNs with L hidden nodes can be obtained as:

$$f(x_j) = \sum_{i=1}^{L} \beta_i g_i(x_j) \quad (1)$$

$$f(x_j) = \sum_{i=1}^{L} \beta_i G(a_i, b_i, x_j) = o_j, j = 1,\ldots,N \quad (2)$$

If the SLFNs can proximate all the N samples without error, that is

$$\sum_{j=1}^{L} || o_j - t_j || = 0 \quad (3)$$

There exist pairs of (a_i, b_i) and β_i such that:

$$\sum_{i=1}^{L} \beta_i G(a_i, b_i, x_i) = t_j, j = 1,\ldots,N \quad (4)$$

The above N equations can also be equivalently expressed in the compact matrix form

$$H\beta = T \quad (5)$$

where

$$H = \begin{bmatrix} G(a_1,b_1,x_1) & \cdots & G(a_L,b_L,x_1) \\ \vdots & \ddots & \vdots \\ G(a_1,b_1,x_N) & \cdots & G(a_L,b_L,x_N) \end{bmatrix}_{NxL} \quad (6)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{Lxm}, \; T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{Nxm} \quad (7)$$

H is called the hidden-layer output matrix of the SLFN. β is represents the output weight matrix, T is the matrix that consists of output labels for the N data samples.

To train a SLFNs as mentioned in (2), it is equivalent to finding the least-square solution $\hat{\beta}$ of linear system (5), that is:

$$\| H\hat{\beta} - T \| = \min_{\beta} \| H\beta - T \| \quad (8)$$

In the case that the number of hidden nodes L is coequal to the number of different training samples N, it is possible to find a $\hat{\beta}$ such that the training error reaches zero. The hidden layer output H is an invertible square matrix. Hence the solution of the linear system can be given as:

$$\hat{\beta} = H^{-1}T \quad (9)$$

In the case that the number of hidden nodes L is less than the number of distinct training samples N, to achieve the smallest training error $\| H\beta - T \|$, the solution of the linear system (5) can be obtained as:

$$\hat{\beta} = H^{\dagger}T \quad (10)$$

Where $H^{\dagger}$ is called the Moore-Penrose universalized inverse [32]. The least squares solution of (8) based on Karush-Kuhn-Tucker (KKT) conditions can be written as

$$\hat{\beta} = H^{\dagger}\left(\frac{1}{C} + HH^T\right)^{-1} T \quad (11)$$

where H is the hidden layer output matrix, C is the regulation coefficient, and T is the expected output matrix of samples. Then, the output function of the ELM learning algorithm is

$$f(x) = h(x)H^{\mathrm{T}}\left(\frac{1}{C} + HH^T\right)^{-1} T \quad (12)$$

If the feature mapping $h(x)$ is unknown and the kernel matrix of ELM based on Mercer's conditions can be defined as follows [33]:

$$M = HH^{\mathrm{T}} : m_{ij} = h(x_i)h(x_j) = k(x_i, x_j), \quad (13)$$

Thus, the output function $f(x)$ of the kernel based ELM can be written compactly as

$$f(x) = [k(x, x_1), \ldots, k(x, x_N)]\left(\frac{1}{C} + M\right)^{-1} T \quad (14)$$

Where $M = HH^{\mathrm{T}}$ and $k(x, y)$ is the kernel function of hidden neurons of single hidden layer feed-forward neural networks. There are many kernel functions satisfying the Mercer's condition available from the existing literature, such as linear kernel, polynomial kernel, Gaussian kernel, and exponential kernel. In this paper, we use Radial Basis (RBF) and Polynomial kernel function for performance analysis.

3. Results and Discussion

12 features were chosen from output data. The chosen features and their explanations are illustrated in (Table.1).

Table 1. Features and explanation

No	Explanation
1	Server port
2	Client port
3	Actual data packets (from client to server)
4	Pushed data packets (from client to server)
5	Pushed data packets (from server to client)
6	Min segment size (from client to server)
7	Average segment size (from server to client)
8	Initial windows bytes (from client to server)
9	Initial windows bytes (from server to client)
10	RTT samples (from client to server)
11	Median data IP(from client to server)
12	Variable data wire (from server to client)

Among the classes belonging to the flows, 7 most commonly used ones were chosen; and 1000 learning processes and 750 tests were applied to each classes. Totally, 7000 classes were chosen for the learning process and 5250 for the test. Chosen classes are illustrated in (Table.2).

Table 2. Classes and data sets

No	Class	Explanation	Training	Test
1	Attack	worm, virus	1000	750
2	P2p	bittorrent,	1000	750
3	Mail	pop3,smtp	1000	750
4	Www	http, https	1000	750
5	Services	dns, ntp	1000	750
6	Bulk	ftp, ssh	1000	750
7	Database	mysql,	1000	750

The success measurement of classification by machine learning algorithms can be examined according to the evaluation table in (Figure.2) including confusion matrix for classification algorithms, and the evaluation metrics [7].

Classified as →	X	$\overline{X}$
X	TP	FN
$\overline{X}$	FP	TN

Figure. 2. Evaluation metrics

In (Figure.2), lines indicate the actual value of the example; and columns of matrix indicate estimated values which were classified or clustered. Accordingly, above-mentioned metrics can be defined as follows:

• True Positive (TP): The number of examples which actually belong to class X and are correctly estimated to be in class X.

• False Positive (FP): The number of examples which don't belong to class X but estimated to be in Class X.

• True Negative (TN): The number of examples which actually don't belong to class X and estimated not to be in class X.

• False Negative (FN): The number of examples which belong to class X but estimated not to be in class X

$$\text{Accuracy}= \frac{TP + TN}{TP + TN + FP + FN} \quad (15)$$

Accuracy rates according to the selected activation function are shown in (Table.3 & 4).

Table 3. Kernel radial basis function

Parameter	Accuracy (%)
100	74.57
10	79.06
1	88.55
0.1	92.38
0.01	95.10
0.001	96.27

Table 4. Kernel polynomial function

Parameter1	Parameter2	Accuracy (%)
1	1	75.14
0.1	0.1	64.88
1	0.1	31.79
0.1	1	75.09
0.01	0.1	60.84
0.001	1	75.45
10	10	92.69
100	100	18.63
1	100	21.14
1	10	93.07
1	100	26.06
100	100	21.31

4. Conclusion

The classification of data on the internet in order to make internet use more efficient has an important place especially for network administrators managing corporate networks. Studies for the classification of internet traffic have increased recently. Machine learning methods of classification of the internet traffic which is commonly used in our day and has been also selected for this study is the kernel based on extreme learning machine based approaches. In this study, over 95% was achieved accuracies using different activation functions. In this study, the kernel based ELM function; Radial Basis and Polynomial functions are used. Functions classification performance by changing the parameters used were observed. RBF is used for the first parameter value. Polynomial function for changing the classification is made and 2 parameter values. With a decrease of the parameter value used for RBF was found that the accuracy increases. This parameter value is 0.01 and reached a value of 95.10% accuracy. With 0,001 of these parameters have reached a value of 96.27% accuracy. But the increase has been observed that a lot of work time. 2 parameter value for Polynomial function is used. The value of 1 and 10 Accuracy rate of 93.07% was observed with the election.

References

[1] K. Xu, F. Wang and L. Gu (2014). Behavior analysis of internet traffic via bipartite graphs and one-mode projections. Networking, IEEE/ACM Transactions on, 22(3), 931– 942.

[2] M. Roughan, S. Sen, O. Spatscheck and N. Duffield (2004). Class-of-service mapping for QoS: a statistical signature-based approach to IP traffic classification, Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, 135 – 148.

[3] X. Zang, A. Tangpong, G. Kesidis and D.J. Miller (2011). Botnet detection through fine flow classification, Departments of CS&E and EE, The Pennsylvania State University, University Park, PA, Report No. CSE11-001.

[4] I. Ismail, M.N. Marsono and S.M. Nor (2010). Detecting worms using data mining techniques: learning in the presence of class noise. In Signal-Image Technology and Internet-Based Systems (SITIS), 187-194.

[5] M. Soysal and E.G. Schmidt (2010). Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. Performance Evaluation, 67(6), 451-467.

[6] T. Karagiannis, K. Papagiannaki and M. Faloutsos (2005). BLINC: multilevel traffic classification in the dark. In ACM SIGCOMM Computer Communication Review 35(4), pp. 229-240.

[7] T.T. Nguyen and G.A. Armitage (2008). Survey of techniques for internet traffic classification using machine learning. Communications Surveys & Tutorials, IEEE, 10(4), pp. 56-76.

[8] A. Callado, C. Kamienski, G. Szabo, B. P. Gero, J. Kelner, S. Fernandes and D.Sadok (2009). A Survey on Internet Identification, IEEE Communications Survey and Tutorials, 11(3), pp 37-52.

[9] IANA, Internet Assigned Numbers Authority [Online] Available: http://www.iana.org/protocols

[10] T. Karagiannis, A. Broido, M. Faloutsos and K. Claffy (2004). Transport Layer Identification of P2P Traffic, IMC'04 Proceeding of the 4th SIGCOMM Conference

on Internet measurement, (ACM New York, New York, U.S.A) pp. 121-134

[11] F. Dehghani, N. Movahhedinia and M. R. Khayyambashi (2010). Real-time Traffic Classification Based on Statistical and Payload Content Features, IEEE ISA, pp. 1-4.

[12] J. Erman, A. Mahanti and M. Arlitt (2006). Internet Traffic Identification Using Machine Learning, IEEE GLOBECOM, pp. 1-6.

[13] L. Yingqiu, L. Wei, L. Yunchun (2007). Network Traffic Classificaion Using K-means Clustering, IEEE IMSCCS, pp. 360-365.

[14] A. W. Moore and K. Papagiannaki (2005). Toward the Accurate Identification of Network Applications, SpringerLink, PAM Lecture Notes in Computer Science, 3431(1), pp. 41-54.

[15] J. Erman, M. Arlitt and A. Mahanti (2006). Traffic Classification Using Clustering Algorithms, MineNet'06 Proceedings of the 2006 SIGCOMM workshop on Mining network data, pp. 281-286.

[16] K. Singh and S. Agrawal (2011). Comparative Analysis of Five Machine Learning Algorithms for IP Traffic Classification, International Conference on Emerging Trends in Networks and Computing Communications (ETNCC), pp. 33-38.

[17] N. Williams, S. Zender and G. Armitage, Evaluating Machine Learning Algorithms for Automated Network Application Identification, Centre for Advanced Internet Architectures (CAIA), Technical Report 060410B, 2006.

[18] N. Williams, S. Zender and G. Armitage (2006). A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification, ACM SIGCOMM Computer Communication Review, 36(5), pp. 5-16.

[19] S. Agrawal and K. Singh (2011). Performance Evaluation of Five Machine Learning Algorithms and Three Feature Selection Algorithms for IP Traffic Classification, IJCA Special Issue on Evolution in Networks and Computer Communications, pp. 25-32.

[20] S. Agrawal and K. Singh (2011). Feature Extraction based IP Traffic Classification using Machine Learning, Proceeding of the International Conference on Advances in Computing and Artificial Intelligence, pp. 208-212.

[21] F. Ertam and E. Avcı (2017). A new approach for internet traffic classification: GA-WK-ELM. Measurement, 95, 135-142.

[22] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule and K. Salamatian (2006). Traffic Classification On The Fly, ACM Special Interest Group on Data Communication Computer Communication Review, 36(2).

[23] G.B. Huang, Q.Y. Zhu and C.K. Siew (2006). Extreme learning machine: theory and applications. Neurocomputing, 70(1).

[24] G.B. Huang and L. Chen (2007). Convex incremental extreme learning machine, Neurocomputing, 70(1), pp.3056–3062.

[25] G.B. Huang and L. Chen (2008). Enhanced random search based incremental extreme learning machine, Neurocomputing, 71(1), pp. 3460–3468.

[26] G.B. Huang, Q.Y. Zhu and C.K. Siew (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. In Neural Networks, Proceedings. IEEE International Joint Conference on 2(1), 985-990.

[27] E. Avci and R. Coteli (2012) A new automatic target recognition system based on wavelet extreme learning machine, Expert Systems with Applications, 39(16), 12340-12348.

[28] G.B. Huang, H. Zhou, X. Ding and R. Zhang (2012). Extreme learning machine for regression and multiclass classification. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 42(2), 513-529.

[29] G.B Huang, X. Ding and H. Zhou (2010). Optimization method based extreme learning machine for classification. Neurocomputing, 74(1), 155-163.

[30] G.B. Huang, Q.Y. Zhu and C.K. Siew (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. In Neural Networks, Proceedings. IEEE International Joint Conference on, 2, pp. 985-990.

[31] J. Luo, C.M. Vong and P.K. Wong (2014). Sparse Bayesian Extreme Learning Machine for Multi-classification, Neural Networks and Learning Systems, 4(25), 836-843.

[32] K.S. Banerjee (1973). Generalized inverse of matrices and its applications, Technometrics, 1(15), 197-197

[33] B. Li, X. Rong and Y. Li (2014). An Improved Kernel Based Extreme Learning Machine for Robot Execution Failures. The Scientific World Journal.

[34] A. Moore, D. Zuev and M. Crogan (2005). Discriminators for use in flow-based classification (Queen Mary and Westfield College, Department of Computer Science)

[35] F. Ertam and E. Avci (2016). Classification with Intelligent Systems for Internet Traffic in Enterprise Networks. Int'l Journal of Computing, Communications & Instrumentation Engg. (IJCCIE) Vol. 3, Issue 1 (2016) ISSN 2349-1469 EISSN 2349-1477