TELKOMNIKA, Vol.12, No.1, March 2014, pp. 201 ~ 208
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v12i1.1051

Research on Network Traffic Identification Based on Multi Layer Perceptron

Dingding Zhou*[1], Wei Liu[2], Wengang Zhou[3], Shi Dong[3]
[1] Department of Laboratory and Equipment Management, Zhoukou Normal University, Zhoukou, Henan,China,466001
[2] Network Management Center, Zhoukou Normal University, Zhoukou, Henan,China,466001
[3] School of Computer Science and Technology, Zhoukou Normal University, Zhoukou, Henan,China,466001
*Corresponding author, email: zdd@zknu.edu.cn

Abstract

In recent years, many machine learning methods have been used in network traffic identification.In order to improve the accuracy and solve some problems of network traffic identification, this paper presents a multi layer perceptron neural network-based method for network traffic identification, and parameters of multi-layer perceptron neural network are analyzed. Experimental results show that this method can effectively solve some problems, and can improve the classification correctness.

Keywords*: Machine learning; Multi layer perceptron neural network; Network traffic identification; Learning rate*

1. Introduction

With the growth of network bandwidth, network traffic identification as a hot research topic in the fields of network management is gradually concerned by researchers at home and abroad. A variety of new network applications such as peer to peer (P2P) are becoming popular, traditional traffic identification method such as well known port based or deep packet inspection are either no longer effective for all type of application. Traffic identification based ML (Machine Learning) can obtain more precise identification accuracy and higher recognition efficiency, and you will need to select appropriate feature selection method, which can select best features according to the impact of the great traffic behavior characteristics. The common traffic identification methods based on machine learning, such as: BAYES neural network, the SVM, C4.5 decision [1-9], these methods are compared in this paper, the results show that the MLP algorithm (Multi Layer Perceptron) has the higher identification accuracy compared with other identification algorithm, and with the increasing of the training samples number, the identification rate has growth trend. In summary, traffic identification algorithm has its own relevance and limitations, this paper proposed a multilayer neural network-based traffic identification algorithm, this algorithm is designed to improve the accuracy of the traffic identification, in-depth analysis of impact of various parameters on the identification results in the multilayer neural network, and provide the reference for further study.

2. Related Work

Since 2004, traffic classification method based on the behavioral characteristics becoming a research focus on the international. Such methods first summarized exhibit different behavioral characteristics between the actual interaction of each application protocol on flow or host, and this application type of traffic will be classified basis for discrimination. It can be divided into two categories with prior training set and the training set, respectively, corresponding to cluster analysis and discriminant analysis in mathematical statistics. Clustering algorithm, A. McGregor, [3] and Jeffrey Erman [4,5] and others respectively use EM and AutoClass clustering method, no training set, consider the similarity between the flow will group flow, and then adopt the port number or inspect packets method to evaluate the accuracy of cluster. Cluster method, however, does not explain why classification flow just so. Therefore, such method can only be used when there is no a priori knowledge on the classification of the training set, a preliminary exploration of the category. Thomas Karagiannis et.al[6] analyze

Received March 15, 2013; Revised October 17, 2013; Accepted January 2, 2014

behavioral characteristics of applications type in the spatial dimension and the features include (port distribution, the number of links, etc.), construct host interaction diagram, and to identify application type between the host. But the method to be cumulative flow, not only there is a lag, and under the high-speed backbone network, how to effective store traffic and rapid construct or match the diagram itself will be still the problem to be solved. M. Roughan [7] and Sebastian Zander [8,9] and others adopt machine learning methods based on k-NN and C4.5, and use the transmission characteristics(flow length , flow duration) to classify the application traffic into 4 to 8 application type in the temporal dimension.Paper [10] provides a framework for concept drifting P2P traffic identification which adopt CluMC algorithm to identify P2P applications.Yaou zhao et.al [11] adopt the ECOC(Error-Correcting Output Codes) based model which was used to improve classification performance.

3. Multi layer perceptron neural network algorithm

Neural network generally consists of input layer, hidden layer and output layer. Each input samples depend on complex nonlinear operation of the network weights and threshold 、 sigmoid function to obtain the output results.

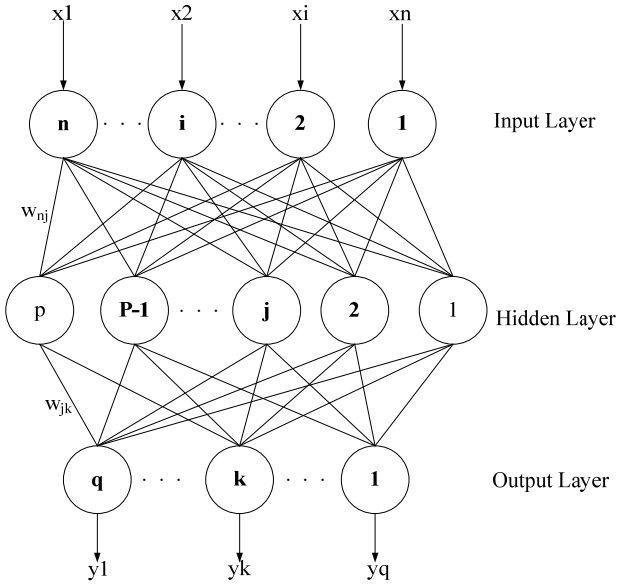

Figure 1. Multi layer perceptron neural network structure diagram

This paper still adopts multi-layer feed forward neural network shown in Figure. 1 to solve problem. Literature [12-15] also uses neural network to identify network traffic application protocol, but identification object is limited to complete TCP flow. Wei L et.al [16] selects 12 features from 248 features by FCBF feature selection algorithm.Shi D et.al [17] proposes improved BP neural network(PCA_BP) method which can improve the identification accuracy.From the literature [18] and analysis of the real trace, we can known such a complete network TCP flows account for only less than 10% of the total flow numbers, obviously, identification object cannot meet the needs of practical engineering and network management, the application protocol identification should be extended to all TCP and UDP traffic on transport layer, including incomplete flow. This extension enables to identify traffic behavior more complicated distribution, the complex degree of the nonlinear boundaries between application types is higher, and greatly increases the noise input of the training process. General neural network training algorithm often cause local minima, the error does not converge, etc. This article further proposes two points on the structure and training of the neural network algorithm.

First, in order to be able to represent the application protocol that may exist between the highly complex nonlinear decision surface and effectively measure the phenomenon of heavy-

tailed distribution, select a sigmoid function as a neuron's excitation function f that output of single neurons which is formula (1).

$$o = \sigma(\vec{w} \bullet \vec{x}) = \frac{1}{1 + e^{-\vec{w} \bullet \vec{x}}} \quad (1)$$

The sigmoid function can map very large input values to a small range of output, so you can measure the great heavy tail distribution phenomena.

Secondly, the paper considers using FR conjugate gradient method (Fletcher-Reeves Conjugate Gradient Method) instead of the traditional gradient descent algorithm [19]. FR conjugate gradient method has faster convergence compared with most conventional gradient descent method, and simply increases the little storage and computation. FR method is not only use of the current gradient as the search direction of error decreases, but to the gradient of the previous point is multiplied by an appropriate coefficient, and is applied to the gradient of the point, thereby obtaining a new search direction. FR conjugate gradient method time step in each orthogonal direction to complete all of them should move in the direction close to the solution x distance, accelerate the understanding of the find speed. FR method iterative equation:

$$\vec{w}_{i+1} = \vec{w}_i + \lambda_i \vec{v}_i \quad (2)$$

Among them, λ_i is called the optimum step size, $E(\vec{w}_i + \lambda_i \vec{v}_i)$ which is the value for the smallest value λ .

$$\vec{v}_i = -\nabla E(\vec{w}_i) + \alpha_i \vec{v}_{i-1}, \quad \alpha_i = \frac{\vec{g}_i^T \vec{g}_i}{\vec{g}_{i-1}^T \vec{g}_{i-1}} = \frac{\left\|\nabla E(\vec{w}_i)\right\|^2}{\left\|\nabla E(\vec{w}_{i-1})\right\|^2},$$

Where, $\nabla E(\vec{w}_i)$ is the partial derivative of the error E for the weight value $\vec{w}$, i.e. gradient.

The neural network training process is carried out by the two-step iteration, the forward propagation and back-propagation.

1: Forward propagation: as shown in Figure.1, the input sample <x1,x2,…,xn> through the hidden layer processing are transmitted to the output layer from the input layer. Current output od and the expected output td are carried out error calculation in the output layer. If the error is less than an acceptable threshold, then the training is successful; Otherwise, go into back propagation process.
2: Back propagation: the error signal output reverse returns according to the original path forward propagation and the formula (2) shown weight coefficient of each hidden layer neuron will be modified by FR conjugate gradient descent method, and then the error signal will tends to a minimum.

The above two steps outlined algorithm is based on the FR conjugate gradient descent method, back-propagation algorithm. Neural network training process that is constantly iteration of these two steps until the error E reaches a predetermined acceptable range, training will be success; or reach a predetermined number of iterations, if the error is still not convergent, then training will be failure.

The monolayer hidden junction FR conjugate gradient descent back propagation algorithm with sigmoid function is shown as follows, in order to highlight the algorithm body, the number of training iterations are no limitation.Training_samples is the form training examples set of ordered pair $< \vec{x}, \vec{t} >$ in algorithm input list. Where $\vec{x}$ is input value vector of the neural network which is corresponding to the behavioral characteristics of the application protocol identification measure value vector, $\vec{t}$ is the target output value of the sample, corresponding to the application protocol type numeral vector; in addition, n_{in} is neural units number of the input layer for the neural network, n_{hidden} is unit number of hidden layer, n_{out} is the number of units of the output layer. Input from unit i to j represents for x_{ji}, weight from unit i to j is expressed as w_{ji}.

Algorithm description:
Algorithm: MLP-based traffic identification algorithm
Input: with the expansion of NETFLOW format flow record $flow_{record}$
Output: with flow record label $flow_{label}$
Algorithm : MLP
Establish neural networks with $\boldsymbol{n_{in}}$ *input units,* $\boldsymbol{n_{hidden}}$ hidden units , $\boldsymbol{n_{out}}$ output units
Initialize all the network weights, $\vec{v} = 0$ 。
do {
for every $< \vec{x}, \vec{t} >$ in *training_samples*
// Step 1. Put $\vec{x}$ into neural network, and calculate output o_u of every unit *u*

$$y = \sum_{k \in Upstream(u)} w_{uk} o_k ,$$

$$o_u = \sigma(y) = \frac{1}{1+e^{-y}}$$

// Step 2. Calculate error δ_k。 Weight gradient and the total error *E* of network output for each output units *k*

$$\delta_k = o_k(1-o_k)(t_k - o_k) ,$$

$$\frac{\partial E}{\partial w_{ki}} = -\delta_k x_{ki} ,$$

$$E(\vec{w}) = E(\vec{w}) + \frac{1}{2}(t_k - o_k)^2 ,$$

// Step 3. Calculate error δ_h and weight gradient for each hidden unit *h*

$$\delta_h = o_h(1-o_h) \sum_{k \in outputs} w_{kh} \delta_k ,$$

$$\frac{\partial E}{\partial w_{hi}} = -\delta_h x_{hi} ,$$

// Step 4. Update each network weight

$$\alpha = \frac{\|\nabla E(\vec{w})\|^2}{\|\nabla E(\vec{w}')\|^2} ,$$

$$\vec{v} = -\nabla E(\vec{w}) + \alpha \vec{v} ,$$

$$\vec{w} = \vec{w} + \lambda \vec{v}$$

// Step 5. Save current weight gradient

$$\nabla E(\vec{w}') = \nabla E(\vec{w})$$

}while($E(\vec{w}) > E_{threshold}$)

4. Experimental and Analysis

4.1. Experimental data

The data set is used in this paper, which is produced by Moore et al [15].They adopt randomly sampling method to produce traffic in the same node from several different time periods in the internet. This node is shared by the three research agencies, which have about 1,000 researchers, technical staff and management, through a full-duplex Gigabit Ethernet link to connect the internet. In 24 hours measuring time, the full-duplex flow through each of the junction are collected by monitoring equipment, so the original data set contains all of the full-duplex flow through the node in 24 hours. Because single data set is too large,Moore,etc. randomly selected ten subsets of data from traffic, in order to verify experimental accuracy and credibility, for the ten extracted data subset, the sampling time of each data set is consistent (both are 28 minutes), and these non-overlapping samples randomly distributed within the 24-hour time interval. Traffic identification based on flow is better able to identify the type of network traffic, and Table 1 lists the common network application type used in the paper, each traffic type contains a different kind of data, such as: Mail type contains smtp, pop3 and other data. Our basic identification object is TCP/IP traffic,it can be expressed as one or more data packets of the communication between two computer using the standard communications protocol (such as: TCP, UDP, ICMP, etc.) in the network,while each network flow is definited by a 5-tuple (source IP address, destination IP address, source port, destination port, transport layer protocol), in experimental classification applications, network flow is divided into uni-directional (one-way) flow and bi-directional(two-way) flow in this paper, in order to focus on the network traffic identification process itself, we only use semantically complete TCP bi-directional flow as a network flow samples available training set constructed and a test set. The complete definition of TCP flow semantics in this paper: observed connection establishment handshake process complete the handshake process (FIN-ACK) TCP (SYN-ACK) and connect the end of the two-way flow.

Table 1. Application type

Classification	Representative applications
WWW	www
MAIL	smtp, pop2/3,imap
BULK	ftp
DATABASE	sqlnet, oracle, ingres, postgres
SERVICE	dns,ident,1dap,ntp,X11
P2P	Bittorrent, Kazaa, Gnutella
INTERACTIVE	ssh, klogin, rlogin, telnet

4.2. Evaluation method

Assessment effectiveness of standard traffic identification algorithm has the following three concepts:

TP (True Positive) is the number of the samples that actually have type i among all those correctly classified as type i by the classification model.

FP (False Positive) is the number of the samples that do not have type i among all those misclassified as type i by the classification model.

FN (False Negative) is the number of the samples that actually have type i among all those classified as another types by the classification model.

Precision

$$\text{Precision}= \frac{TP}{TP+FP} \tag{3}$$

Recall

$$\text{Recall}= \frac{TP}{TP+FN} \tag{4}$$

Overall accuracy

$$\text{Overall accuracy}= \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FP_i)} \quad (5)$$

4.3. Experimental results and analysis

In order to verify and analyze the stability and effectiveness of the MLP algorithm, 10-fold cross-validation is used to compare precision, recall and overall accuracy shown in formula (3) (4) (5) . this paper analyzes the impact of the learning rate and sample size on this algorithm, and other classic classification algorithm.

Analysis from Figure 2, when the learning rate is 0.2, the highest is overall accuracy.

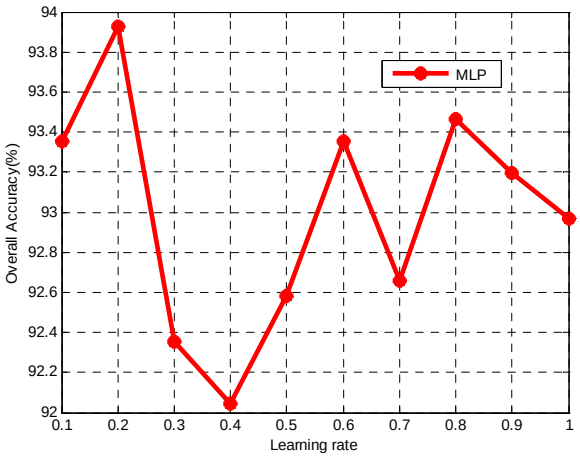

Figure 2. The impact of learning rate on overall accuracy

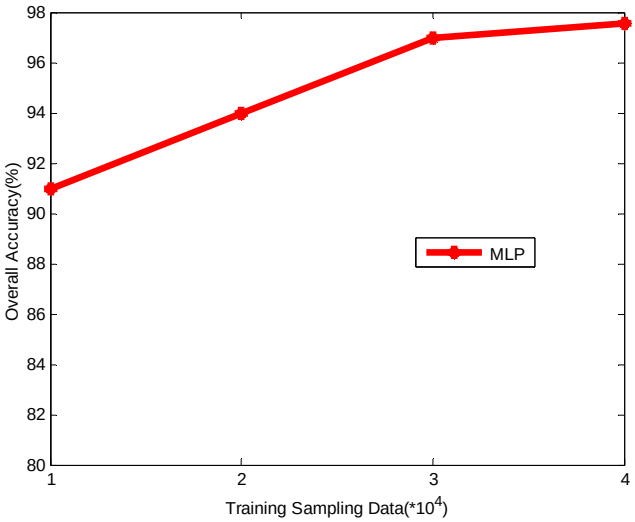

Figure 3. The impact of training sampling on overall accuracy

From Figure 3, the results show that with the increasing of the training sampling data, overall accuracy has upward trend, and it increases more significantly from $1*10^4$ to $3*10^4$, rises gradually slowly from $3*10^4$ to $4*10^4$. It proves that size of training sampling data has certain effect on overall accuracy, especially when the size is small, which mainly is due to error reducing from weight adjusting. While the bigger sampling data will be smaller weight adjusting, so the error will be increasing and overall accuracy of traffic identification rises gradually slowly.

Table 2. Precision of four algorithms

Application	PRECITION			
	MLP	BAYES	C4.5	SVM
WWW	98.63%	96.34%	97.28%	97.12%
MAIL	98.16%	96.2%	96.83%	96.46%
BULK	97.97%	96.13%	96.25%	96.36%
DATABASE	96.29%	95.83%	95.96%	96.12%
SERVICE	93.36%	92.18%	93.12%	92.89%
P2P	94.88%	93.89%	94.14%	94.26%
INTER	99.12%	98.21%	98.76%	98.89%

Table 3. Recall of four algorithms

Application	RECALL			
	MLP	BAYES	C4.5	SVM
WWW	97.98%	95.86%	96.12%	96.43%
MAIL	97.83%	95.46%	95.87%	95.94%
BULK	97.35%	95.71%	96.13%	96.54%
DATABASE	96.11%	95.42%	95.77%	95.87%
SERVICE	92.79%	92.13%	92.34%	92.58%
P2P	95.36%	94.28%	94.78%	94.82%
INTER	98.79%	97.93%	98.12%	98.46%

From Table 2 and Table 3, the experimental results show that, either precision or recall rate MLP have gained the highest.The reason why is that MLP can adopt the minimization error to evaluate the objective function which will not consider the singer the network application. Other algorithms have preference [20] on traditional algorithm (BAYES, C4.5, SVM),so which will be inevitably influenced.

Table 4 Overall Accuracy four algorithms

Algorithm	Overall Accuracy
MLP	97.56%
BAYES	92.34%
C4.5	96.75%
SVM	95.89%

As shown in Table 4, MLP has highest overall accuracy compared with Bayes, C4.5, and SVM. The reason why is that MLP can overcome the impact of noise input data and adopts the appropriate learning rate to improve the identification accuracy.

5. Conclusion

In this paper, a traffic identification algorithm based on multi-layer perceptron neural network (MLP algorithm) is proposed, and the impact of the neural network parameters on MLP algorithm is analyzed and studied. The experimental results show that: the learning rate and sample number both have great effect on MLP algorithm, MLP has higher precision and identification accuracy compared with other algorithms.

Based on this study, the next step research work is: the impact of sampling method on MLP algorithm will be analyzed, and MLP algorithm will be applied to the real network environment with a sampling strategy.

Acknowledgements

This paper is supported by Program for Science and Technology Development of department of science and Technology in Henan Province(102102210265) and Program for Basic and cutting-edge of department of science and Technology in Henan Province (132300410276).

References

[1] Andrew W. Moore, Konstantina Papagiannaki. *Toward the Accurate Identification of Network Applications*. In: Proc. of PAM 2005. Boston, USA. 2005: 41-54.
[2] Myung-Sup Kim, Young J. Won, James Won-Ki Hong. Application-Level Traffic Monitoring and an Analysis on IP Networks. *ETRI journal*. 2005; 27(11): 22-42.
[3] A. McGregor, M. Hall, P. Lorier, J. Brunskill.*Flow Clustering Using Machine Learning Techniques*. In: Proc. of PAM 2004. Antibes Juan-les-Pins, France. 2004: 205-214.
[4] Jeffrey Erman, Martin Arlitt, Anirban Mahanti.*Traffic Classification Using Clustering Algorithms*. In: Proc. of ACM SIGCOMM Workshop on Mining Network Data 2006. Pisa, Italy, 2006: 281-286.
[5] Jeffrey Erman, Anirban Mahanti, Martin Arlitt. *Internet Traffic Identification using Machine Learning*. In: Proc. of 49th IEEE Global Telecommunications Conference. San Francisco, USA. 2006: 1-6.
[6] Thomas Karagiannis, Konstantina Papagiannaki, Michalis Faloutsos. *BLINC: Multilevel Traffic Classification in the Dark*. In: Proc. of ACM SIGCOMM 2005. Philadelphia, USA. 2005: 229-240.
[7] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield. *Class-of-service mapping for QoS: A Statistical Signature-based Approach to IP Traffic Classification*. In: Proc. of ACM SIGCOMM IMC 2004. Taormina, Italy. 2004: 135-148.
[8] Sebastian Zander, Thuy Nguyen, Grenville J. Armitage. *Self-Learning IP Traffic Classification Based on Statistical Flow Characteristics*. In: Proc. of PAM2005. Boston, USA. 2005: 325-328.
[9] Sebastian Zander, Nigel Williams, Grenville Armitage. *Internet Archeology: Estimating Individual Application Trends in Incomplete Historic Traffic Traces*. In: Proc. of PAM 2006. Adelaide, Australia. 2006: 205-206.
[10] Guanghui Yan, Minghao Ai.A framework for concept drifiting P2P traffic identification. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(8): 4317-4326.
[11] Yaou zhao, xiao xie, mingyan jiang.Hierachical realtime network traffic classification based on ECOC.*TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2014; 12(2).
[12] Denis Zuev, Andrew W. Moore. *Traffic Classification using a Statistical Approach*. In: Proc. of the 6th annual Passive and Active Measurements Workshop (PAM'05). Boston, USA. 2005: 321-324.
[13] Andrew W. Moore, Denis Zuev.*Internet Traffic Classification Using Bayesian Analysis Techniques.* In: Proc. of ACM SIGMETRICS'05. Banff, Canada. 2005: 50-60.
[14] Hongbo Jiang, Andrew W. Moore, Zihui Ge, Shudong Jin, Jia Wang.*Lightweight Application Classification for Network Management.* In: Proc. of the SIGCOMM Workshop on Internet Network Management'07. Kyoto, Japan. 2007: 299–304.
[15] T Auld, AW Moore, SF Gull.Bayesian Neural Networks for Internet Traffic Classification. *IEEE Transactions on Neural Networks*. 2007; 18(1): 223-239.
[16] Wei Li, Marco Canini, Andrew W. Moore, Raffaele Bolla.Efficient application identification and the temporal and spatial stability of classification schema.*Computer Networks.* 2009; 53(6): 790–809.
[17] Shi Dong, Dingding Zhou, Wengang Zhou, et.al. Research on network traffic identification based on improved BP neural network.*Applied Mathematics & Information Science*. 2013; 7(1): 389-398.
[18] Mingzhong Zhou. Study of Large-scale Network IP Flows Behavior Characteristics and Measurement Algorithm. PhD Thesis.Southeast University; 2006.
[19] Satish Kumar. Neural Networks. USA: McGraw-Hill. 2004.
[20] Shi Dong, Dingding Zhou, Wei Ding, et.al. Traffic classification model based on integration of multiple classifiers. *Journal of Computational Information Systems.* 2012; 8(24): 10429-10437.