Advances in Computer Science Research, volume 44
3rd International Conference on Wireless Communication and Sensor Network (WCSN 2016)

A Web Page Classification Method Based on TCP/IP Header Features

Di Huang, Xin-Yi Zhang, Qi-Wei Tang
Shanghai Jiao Tong University, Shanghai, China
E-mail: 2434520266@qq.com, 283802073@qq.com, 1421940251@qq.com

***Abstract*-Web page classification has wide applications. Due to various types of web pages and vast amounts of network traffic, it is difficult to classify web pages by deeply inspecting the content of each packet. This paper presents a learning-based classification method according to TCP/IP header features. First, we propose an approach to select features and improve the Relief algorithm, which can pick features with robustness. Then we raise a labeling strategy to assign each feature with a label when training the classifier. Last, we put forward a learning-based classification method which takes labels and multi-layer semantics into consideration. The experiment results show that the proposed strategy can improve the processing speed and the accuracy of classification.**

Keywords- web page classification; packet header feature; instance-based learning

I INTRODUCTION

Web is the widest application in networks and HTTP messages account for approximately 80% of network traffic [1]. Web page classification has applications in multiple areas, such as intrusion detection, targeted advertising and traffic modeling. The ideal method of web page classification is classifying basing on their contents, however, some methods can only be applied to text classification while some are limited to particular languages [2]. As networks grow in size, the genres of web pages become more complicated, including pictures, audio and videos besides text. Researchers have proposed improved ways to make classification methods accommodate to the network development [3].

In recent years, web page classification techniques have made much progress. But most of them obtain feature information for classification by deeply inspecting the content of each packet. Therefore, they need to monitor access connection and the network traffic created by interaction between client and server [4]. As for network traffic classification, researchers mainly focus on Application-level protocol (such as FTP and HTTP) classification [5]. However, little attention is paid to classifying web pages according to genres, which is quite challenging [6].

This paper proposes an approach to classify web pages based on TCP/IP header information in network traffic. The main contribution is: (1)Define labels for web pages and design different standards for classification; (2)Propose an improved way to select web page features, which can pick the feature with robustness and maintain relative independence from the network environment; (3)Design a learning-based classification method to achieve high accuracy for classification.

II RELATED WORK

The design of TCP header is to ensure more accurate data transmission in networks. The option field in TCP header can not only adapt itself to the complex and capricious network, but also can serve the application layer better [7][8]. TCP header can also be used to identify operating systems [9]. By analyzing data options in the packet and other features with Bayes Rule [10], we can identify the host operating system who sends these packets.

TT Yao, et al. used the Support Vector Machine(SVM) to classify web pages [11]. However, SVM is a binary classifier, which has a great limitation. K-Nearest Neighbor(KNN) [12] is an algorithm used for classification and regression. In recent years, it has been used for web page classification.

In [13] the authors improve web page classification precision by using two-level key words and hierarchical classification method. And in [14], Multi-Neighbor Attribute Classification is used to increase the accuracy of link-based classification.

Traditional classification methods are mostly based on text processing, and there are many shortcomings including: (1) Depending too much on text contents. If the web page provider inserts false information for some purpose, then traditional classification methods cannot identify those useless information. Meanwhile, these methods are not applicable to audios, videos or pictures. (2) Closely related to language. For example, a method aimed to classify Chinese web pages is not necessarily applicable to English pages. (3) High dimension in vector space model. Therefore, it is difficult to achieve a balance between efficiency and accuracy.

III MULTIPLE-INSTANCE LEARNING FOR CLASSIFICATION

A Definition

1) *Feature selection based on TCP/IP header*

TCP/IP header contains information about time, multi-flow feature and extended statistics feature, but only part of them can provide useful information. Due to dynamic variation of networks, some features are characterized as dynamic, causing changes in the accuracy of classification. Hence, the primary problem is selecting features that are effective for web page classification.

2) *Web page label definition*

With massive amounts of information in networks, applying the classification algorithm directly can increase the difficulty of classification. However, defining

appropriate labels for web pages can reduce the difficulty and the amount of computation. Similar to keyword, label is used to mark the category or the content of a specific feature. Classification based on label can easily descript and retrieve relevant content. Moreover, well defined labels should satisfy diversified demands under different scenarios.

3) *Fast classification algorithm*

Classification algorithm should possess high performance and high accuracy for huge amounts of web pages. Meanwhile, it should be applicable to different application scenarios.

B Classification Algorithm Overview

As indicated by Figure 1, the classification algorithm is mainly composed of several parts: (1)Data flow tracing: analyze web pages to trace data flows generated by different TCP connections. (2) Feature extraction: extract important features by implementing real-time analysis of web pages. (3) Feature selection: select effective features based on relief algorithm and the average deviation method. (4) Label strategy: define labels and mark training data to train the classifier. (5) Web page classification: classification method is based on machine learning, and we call it Multi-Instance Multi-Label learning.

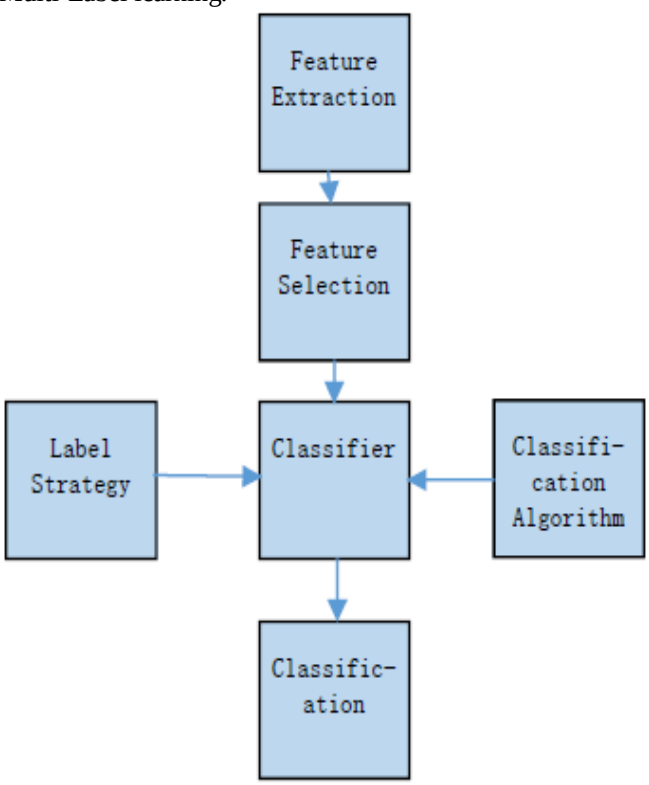

Figure 1. Classification method framework.

This paper uses features from TCP/IP header, so we need to trace data flow and extract all features from TCP/IP header. Features that can be extracted include the number of PUSH labels in the TCP connection, the transmission size of HTTP objects, the number of TCP connections generated by a web page, round-triple time(RTT) and transmission delay. Additionally, we can get statistical characteristics such as the maximum value, the minimum value and the percentage.

C Feature Selection

The principles of feature selection are: (1) select features that provide the most effective information for web page classification; (2) group those effective features into highly relevant feature subsets; (3) pick the most stable (e.g. no significant change over time) and most consistent (e.g. consistent with different browsers) feature from each subset.

Let Mn,b,t denote the measurement model for feature i in n×b×t web page space, where n denotes the number of web pages, b denotes the number of browsers and t denotes the number of web pages reloaded over time. We group the selected features into related subsets. Here, we use the Pearson correlation coefficient ρ to identify several groups of features that are highly correlated, namely, ρ≥0.75 for features in the same group while features from different groups have coefficient ρ<0.3. The distance between features is the maximum distance when the category of sample is unchanged, which can be indicated as:

$$\Theta = (|x\text{-M}(x)| - |x\text{-H}(x)|)/2 \quad (1)$$

H(x) denotes the nearest neighbor of x which has the same type as x and M(x) denotes the nearest one having the different type from x. These values can evaluate the classification ability of features in any dimension. To be more specific, suppose X={x1,x2,...xn} denotes all the objects to be classified and xi=[xi1,xi2,... xiN]T denotes N features of the i-th sample. We randomly select m samples from the training data, compute distances between samples of all the features (denoted by diff(i,x,H(x))/m and diff(i,x,M(x))/m) and calculate the sum of them as the weight for features, shown by formula (2). From this formula we know that when features are easy to be classified, the distance between samples of the same type is small while the distance between samples of different types is big.

$$\text{Wi+1=Wi-diff(i,x,H(x))/m+diff(i,x,M(x))/m} \quad (2)$$

D Feature Stability And Feature Consistency

The ideal features should stay relatively stable over time, namely, their values should not change significantly. The stability is defined by the average deviation ratio. For each feature i, a n×t model is defined by: $S_{n,t}^{i} = \sum_{b} \omega(b) M_{n,b,t}^{i}$, where ω(b)∈[0,1] represents the value in browser b. The stability of each feature i and web page n over time, namely, the average deviation ratio is defined by formula (3), where $\mu_n^i = \sum_t S_{n,t}^i / T$.

$$DS_n^i = 100 \sum_{t=1}^{T} | S_{n,t}^i - \mu_n^i | / T\mu_n^i \quad (3)$$

$$DC_n^i = 100\sum_{b=1}^{B}\omega(b)\,|\,c_{n,b}^i - v_n^i\,|\,/\,v_n^i \quad (4)$$

The consistency of feature i and web page n in different browsers is defined by the corresponding average deviation ratio, as shown in formula (4), where $c_{n,b}^i = \sum_t M_{n,b,t}^i / T$ is an n×b model. When browser b reloads web page n, each element in it represents the average measurement of feature i, and $v_n^i = \sum_b \omega(b) C_{n,b}^i$.

IV FEATURE SELECTION AND CLASSIFICATION ALGORITHM

A Feature Selection Algorithm

Let D denote the training data set, n denote the number of sampling times, δ denote the range of feature weight and T denote the weight of each feature. We design the feature selection algorithm as following:

Algorithm 1: feature selection algorithm
Input: D, n, m, δ
Output: T

1. Reset 0 to all features and T=φ
2. for i=1 to m do
3. Randomly select a sample R;
4. find the nearest neighbor H of R From similar samples, find the nearest neighbor sample M from different kinds of samples
5. for A=1 to n do
6. W(A) := W(A)-diff(A,R,H)/m+diff(A,R,M)/m
7. for A=1 to n do
8. If W(A) ≥ δ
9. Add the first feature to T
10. Divide the features into 10 groups
11. Let $DS_n^i := 100\sum_{t=1}^{T}|\,S_{n,t}^i - \mu_n^i\,|\,/\,T\mu_n^i$
12. $DC_n^i := 100\sum_{b=1}^{B}\omega(b)\,|\,c_{n,b}^i - v_n^i\,|\,/\,v_n^i$
13. choose middle (DS_n^i) and middle (DC_n^i) from each group

B Web Page Classification Algorithm

We define four labels for web page classification according to contents and types of pages: (1)VSL label: including video stream web pages and non-video stream web pages; (2)TDL label: including traditional pages and mobile device pages; (3)AGL label: web page type based on Alexa ranking, including pages about news, shopping or business; (4)WNL label: including login pages, searching pages and other pages that contain some clickable contents. Every web page will be labeled and the label correspond to a type.

We adopt a Multiple-Instance and Multiple-Learning (MIML) frame. The MIML frame deals with the diversity in input space as well as output space. What it learns is the way to map elements from the instance space to the classification label set. Let X denote the instance space and Y denote the label set, then for a given data set {(X1, Y1), (X2, Y2),...,(Xm, Ym)}, the goal is to find f: 2x→2y, where Xi⊆X is a set of instances $\{x_{i1}, x_{i2}, \ldots x_{in_i}\}$, $x_{ij} \in X$ (where j = 1, 2, ... , ni), and Yi⊆Y is a set of appropriate labels $\{y_{i1}, y_{i2}, \ldots y_{il_i}\}$ for Xi, where $y_{ik} \in Y$ (k = 1, 2, ...,li). Notice that ni is the number of instances that Xi contains and li is the number of labels that Yi contains. We define the MIML classification algorithm as following:

Algorithm 2: MIML Classification algorithm

1. Converting the MIML sample (X_u,Y_u) (u= 1,2,...,m) to $|Y|$ instance packets $\{[(X_u, y_1), f_{\mathrm{MIL}}(X_u, y_1)],..., [(X_u, y_{|Y|}), f_{MIL}(X_u, y_{|Y|})]\}$, then the original data sets are transformed into multiple-instance data sets which contain $m*|Y|$ instance packets, denoted by $[(X^{(i)}, y^{(i)}), f_{MIL}(X^{(i)}, y^{(i)})]$ (i=1, 2,..., $m*|Y|$).
2. Set the initial value of every packet's weight to be $W^{(i)}=1/(m*|Y|)$ (i=1, 2,..., $m*|Y|$).
3. for t = 1, 2, ..., T:
4. { Set $W_j^{(i)} = W^{(i)}/n_i$ (i = 1, 2, ..., $m*|Y|$), assign the labels $f_{MIL}(X^{(i)}, y^{(i)})$ to every instance $(X^{(i)}, y^{(i)})$ (i=1, 2, ..., n_i), and then create an indicator $h_t[(X_j^{(i)}, y^{(i)})] \in \{-1, +1\}$.
5. For the i-th packet of each group, figure out the error rate $e^{(i)} \in [0,1]$ by calculating the number of classification errors in packets: $e^{(i)} = \sum_{j=1}^{n_i}[|\,h_t[x_j^{(i)}, y^{(i)}] \neq f_{MIL}(x^{(i)}, y^{(i)})\,|]/n_i$.
6. If e(i)<0.5 for all i∈{1,2,...,m*|Y|}, go to step 10.
7. $c_t := \arg\min \sum_{i=1}^{m*|Y|} W^{(i)} \exp[2e^{(i)} - 1]$ (namely, ct makes the expression in the right reach the minimum)
7. If $c_t \leq 0$, go to step 10.
8. Set $W^{(i)}=W^{(i)}\exp[(2e^{(i)}-1)_{\mathrm{ct}}]$ (i=1,2,...$m*|Y|$) and then normalize it so that $0 \leq W^{(i)} \leq 1$ and $\sum_{i=1}^{m*|Y|} W^{(i)} = 1$ } (end of "for" in step 3)
9. $Y^* := \{y \,|\, \arg_{y \in Y} sign(\sum_j \sum_t c_t h_t[(x_j^*, y)]) = +1\}$ (where x_j^* is the j-th instance of X^*) and return Y^*

V EXPERIMENTS AND ANALYSIS

Simulation data is grabbed from Internet in a certain period of time. We use a part of it as training data, and the other part of it is used for testing. The experiment is implemented with MATLAB.

A Effects of Labeling Strategy

The result of labeling strategy experiment is shown in Figure 2, where the vertical axis represents the accuracy. According to [13], generally we should achieve the classification accuracy of more than 70%. From Figure 2 we can find that different classes of web pages have different classification accuracy. The recognition rate of video web pages (VSL) is 99%, while it is 90% for mobile web pages (TDL), 82% for navigation pages(WNL) and 73% for pages classified by genre(AGL). This is because video web pages

have more obvious features.

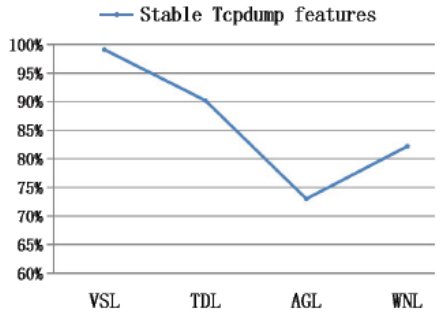

Figure 2. Web page classification accuracy with labeling strategy.

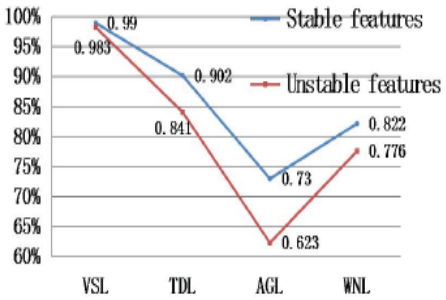

Figure 3. Web page classification accuracy of stable features and unstable features.

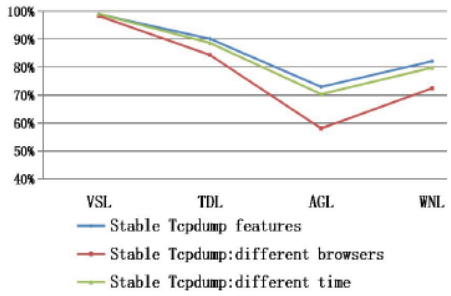

Figure 4. Sensitivity to time and browsers.

B *Effects of Feature Stability*

By loading web page several times in different time span, we test the effect of feature stability on web page classification accuracy. Figure 3 shows that web pages classified by stable features have higher classification accuracy.

C *Sensitivity to Time and Browser Platform*

Figure 4 shows the result of reloading each web page six times with five different browsers. According to the result, video web pages have the same recognition rate while the recognition rate of mobile web pages and navigation web pages is 6 to 10 percent lower. For labels based on genre, the accuracy is only about 58%.

VI CONCLUSION

This paper improves the existing classification standard for web pages, defines a variety of labels and designs different standards. We also optimize the web page feature selecting method basing on those proposed standards and make it easier to find out the features with stability, namely, the features that are independent of the network environment and would not change significantly when the network environment changes. The proposed learning algorithm has high accuracy of classification. However, the proposed algorithm has not taken full advantage of the information of multiple streams, on which further research should focus.

REFERENCES

[1] Meisam Eslahi, M. S. Rohmad, Hamid Nilsaz, Maryam Var Naseri, N. M. Tahir, H. Hashim. "Periodicity classification of HTTP traffic to detect HTTP Botnets." 2015 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE) , 2015, pp. 119 – 123.

[2] Zhou Hongfang. "A Web Page Classification Algorithm Based on Feature Selection." Journal of Information & Computational Science,12.4(2015):1549-1556.

[3] Chouhan Jayendra Singh, A. Gadwal. "Improving web search user query relevance using content based page rank." 2015 International Conference on Computer, Communication and Control (IC4), 2015, pp.1-5.

[4] Justine Sherry, Chang Lan, Raluca Ada Popa, Sylvia Ratnasamy. "BlindBox: Deep Packet Inspection over Encrypted Traffic." ACM SIGCOMM Computer Communication Review,45.5(2015):213-226.

[5] Lu Wei, L. Xue. "A Heuristic-Based Co-clustering Algorithm for the Internet Traffic Classification." International Conference on Advanced Information NETWORKING and Applications Workshops 2014:49-54.

[6] S.Sanders, J.Kaur. "Can web pages be classified using anonymized TCP/IP headers?" 2015 IEEE International Conference on Computer Communications (INFOCOM), 2015, pp.2272 – 2280.

[7] A.Osanaiye, M. Dlodlo. "TCP/IP header classification for detecting spoofed DDoS attack in Cloud environment." Eurocon 2015 - International Conference on Computer As A Tool, 2015, pp.1-6.

[8] D.Naylor, P. Steenkiste. "Do You Know Where Your Headers Are? Comparing the Privacy of Network Architectures with Share Count Analysis." Proceedings of the 14th ACM Workshop on Hot Topics in Networks, 2015.

[9] YC Chen, Y Liao, M Baldi, SJ Lee, L Qiu. "OS Fingerprinting and Tethering Detection in Mobile Networks." Proceedings of the 2014 Conference on Internet Measurement Conference, 2014.

[10] Allan E. Clark, R. Altwegg, John T. Ormerod. "A Variational Bayes Approach to the Analysis of Occupancy Models." Plos One 11.2(2016):1-18.

[11] TT Yao, JL Cheng, BR Xu, MZ Zhang, YZ Hu. "Support vector machine (SVM) classification model based rational design of novel tetronic acid derivatives as potent insecticidal and acaricidal agents." Rsc Advances 5(2015).

[12] Leif Peterson. "K-nearest neighbor." Scholarpedia, 2009,4(2).

[13] Andrey N. Rukavitsyn, Mikhail S. Kupriyanov, Andrey V. Shorov and Ilya V. Petukhov,"Investigation of Website Classification Methods Based on Data Mining Techniques," IEEE International Conference on Soft Computing and Measurements (SCM), 2016, pp. 333-336.

[14] Luke K. Mcdowell and David W. Aha, "Leveraging Neighbor Attributes for Classification in Sparsely Labeled Networks," ACM Transactions on Knowledge Discovery from Data (TKDD),vol. 11,No. 1, Article 2, pp.1-37. July 2016.