Web Robot Detection based on Monotonous Behavior

Shinil Kwon[1], Myeongjin Oh[1], Dukyun Kim[1], Junsup Lee[2], Young-Gab Kim[1], and Sungdeok Cha[1, *]

[1]College of Information and Communication, Korea University, Anam-dong Seongbuk-Gu, Seoul, 136-701 South Korea
{shinilkwon1, iamohmj, kim9069}@gmail.com, {always, scha}@korea.ac.kr
[2]Convergent Contents Research Department, Electronics and Telecommunication Research Institute, 161 Gajeong-dong, Yuseong-gu, Daejeon, 305-700, South Korea
junsup@etri.re.kr

Abstract. Several studies examined various features on how to most effectively detect web robots. Based on an insight that most web robots, regardless of specifics, would exhibit focused and therefore monotonous behavior, this paper proposes that monitoring the rate of behavioral change is highly effective in detecting sessions initiated by web robots. Empirical evaluation performed on more than one billion requests made to www.microsoft.com web servers confirms that "switching factor" is indeed highly effective. In this paper, we explain the three features whose switching factor was used in web robot detection. Unlike previous studies where the types of web robots to detect were limited to a few types (e.g., text crawlers, link checkers, and email harvester), we extend the types of web robots to image collectors, icon collectors, download tools, etc.

Keywords: web robot detection, classification, data mining, switching factor

1 Introduction

Accurate and rapid detection of various web robots has become an important research topic with the explosive growth of the web-based economy. In addition to text crawlers deployed by search engine companies, there are a variety of web robots such as paper crawlers, link checkers, email harvesters, and icon crawlers. Some hostile web robots pose serious security threats. One must be able to distinguish sessions initiated by interactive users surfing the web from those created by programmed web robots.

D. Doran and S. Gokhale [1] classified existing robot detection techniques into four categories: syntactical log analysis, analytical learning techniques, traffic pattern analysis, and Turing test systems. Among these categories, analytical learning techniques and traffic pattern analysis closely relate to the anomaly detection used by this study. The analytical learning techniques focused on finding various features of each web robot (e.g., 26 or 34 features) with a running average or standard deviation, and

* Corresponding author

adfa, p. 1, 2012.

finally, concentrate on web robot detection using a data mining algorithm. Tan & Kumar [2] examined web robot detection in terms of 26 features such as the ratio of empty strings in the referrer field and the ratio of the head method with the C4.5 decision tree algorithm. Lourenco and Belo [3] chose roughly same features and C4.5 decision tree algorithm to Tan and Kumar. Bomhardt et al.[4] added eight features and applied a neural network using web log preprocessing tool called. Stassopoulou and Dikaiakos [5] constructed a Bayesian network using six features such as Maximum sustained click rate, Duration of session. Lu and Yu [6] detected web robots using hidden Markov model (HMM) and inter arrival time.

The traffic pattern analysis proposed new features based on behavioral analysis of web robots. Almeida et al. [7] presented a characterization of the workload generated by web robots. This characterization focuses on the statistical properties of the arrival process and on the robot behavior graph model. Huntington et al. [8] proposed a multi-step log analysis technique with online scientific journal Glycobiology. Duskin and Feitelson [9] studied the interaction between the query submittal rate and the minimal interval of time between different queries. Ye et al. [10] measured a server workload using difference of HTTP response time and ICMP response time.

Web robot detection using a switching factor, as reported in this paper, overcomes limitations of other studies: our approach detected each web robot with equal performance using our new features. Although the six different types of web robots had great imbalance in the number of sessions, we successively detected each web robot without discarding the text crawler sessions by using only three features common to all web robots.

The remainder of our paper is organized as follows. In Section 2, we introduce the concept of switching factors, with an illustrative example. In Section 3, we explain the Microsoft web log used in this study. Section 4 details experimental results and compares our results against those reported by Tan and Kumar. Section 5 concludes the paper.

2 Switching factor

Our approach is based on an insight that all web robots, regardless of their specific tasks they have been programmed to perform, would exhibit highly focused and therefore consistent behavior unlike that exhibited by interactive users. Web robots may employ diverse search strategies (e.g., depth-first vs. breath-first), algorithms, or data structures. For example, we discovered in Microsoft log that some sessions initiated by an image crawler seldom contained image requests. While strange at first, further analysis revealed that image requests were made by different and spawned threads. Different image crawlers left traces indicating that crawling and image requests were all made in the same thread. Because such image crawlers exhibited vastly different in strategies, direct comparison of features may yield little similarities. However, they exhibited highly focused and consistent behavior, and “switching factor” for both web robots turned out to be extremely small when compared to the web sessions generated by interactive users.

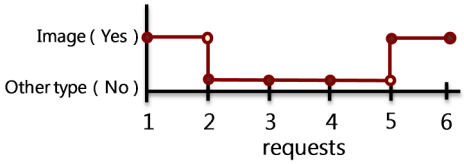

Fig. 1. Is there a request to an image in the log?

Figure 1 illustrates the switching factor concept. It illustrates a session in which image requests occur in the first, fifth, and sixth requests. Rather than counting the ratio of image requests (e.g., 3 out of 6 in this example), as [2] did, we observe that "behavioral changes" occurred at the second and fifth requests. For a session with N requests, there can be at most (N-1) opportunities for changes. The switching factor in this example is 0.4 (or 2 out of 5) as defined below:

Definition 1 For each feature F, switching factor (SF) is the ratio of changes between all the adjacent requests.

$$SF(F) = \frac{\text{Number of requests whose F changed from the preceding request}}{\text{Number of requests} - 1}$$

3 A proposed approach to web robot detection

3.1 Web server log www.microsoft.com

The web server log used in this study is same as that used in [11]. It is about 250 GB. More than one billion requests are captured in 24 hours. We divided the web log into four segments, each containing six hours, to empirically evaluate web robot detection accuracy. Care was taken to ensure that "peak" hours in different continents (e.g., North America, Asia, and Europe) are properly reflected in each segment. Data belonging to each segment is further divided into training and test data in a 2:1 ratio. That is, logs collected during the first four hours were used as training data, and the rest were used as test data.

3.2 Selection of detection features

We selected three fields in the IIS format log as initial candidates based on an earlier study [11], built databases containing sessions initiated by interactive users as well as web robots, computed switching factors on each features.

Switching factor on unassigned referrer field (cs-referrer).
Conventional wisdom on web robots is that referrer field would contain empty string (e.g., unassigned). While generally true, our analysis on Microsoft log revealed exceptions. Girafa and Arachmo had string value included in the referrer field in more than 80% of the requests. All other web robots almost always had unassigned referrer

field. That is why inclusion of empty string is not always an effective, though straightforward, feature. However, switching factors are consistently

Switching factor of file types (cs-uri-stem).
While Microsoft logs consist of various types of file such as web pages or video files, almost all their requests are web page and image files. (e.g., Web page: 70.5%, Image file: 29%) Interestingly, the requesting patterns in these two file types are entirely different. With the apparent exception of image or icon crawlers, web robots are less likely to be interested in images. Text crawlers and link checkers are the most obvious samples. Among the text crawlers, types of requested resource varied depending on who launched them (e.g., Google vs. Yahoo). For interactive users who generally traverse pages by clicking the links, actual request would be generated by a browser. In such cases, image requests would be mixed. In fact, interactive users rarely requested only images (i.e., 1.9%), and such superficial criteria would be ineffective in separating behavior of text crawlers from that of interactive users. However, switching factors on web robots are unanimously and significantly below that of interactive users. One must ask whether or not behavioral patterns change frequently between the adjacent requests because switching factor analysis on web robots returned unanimously and significantly lower values compared to that of interactive users.

Switching factor on number of bytes from clients to the server (cs-bytes).
Analysis on the number of bytes clients sent to the server revealed similar pattern to other features. Lee et al. [11] reported that majority of web robots, except the image crawler named Girafa, sent smaller number of bytes (e.g., 320 bytes) than interactive users did (e.g., 490 bytes) in almost all requests (e.g., 95.57% to 100%). About 8% of Girafa requests, however, did not follow the pattern. While generally useful, such sporadic exceptions may result in false alarms in web robot detection. We instead compare the current request size against the running average of all the requests and ask if there are changes in patterns (e.g., below or above the average). Such analysis resulted in consistent and substantially lower values when applied on web robots. This is yet another evidence of highly monotonous behavior of web robots.

4 Experimental evaluation

F-measure, computed from recall and precision values, is widely used in machine learning research. It provides an impartial viewpoint in that one can increase the value of one at the expense of the other. On one extreme, an algorithm may achieve 100% recall value if it simply declares all the sessions as those initiated by web robots. Such algorithm is obviously useless and filled with numerous false alarms.

In running the experiment, we used an open-source implementation of machine learning algorithm commonly known as C4.5 [12]. Many research groups working on web robot detection, including Tan and Kumar, used the same software. We changed session lengths (e.g., the minimum number of requests included in a valid session)

from three to eight, repeatedly measured precision and recall values for the chosen features. For example, if session length is set to eight, all the sessions containing seven or less requests are excluded from the training and test data.

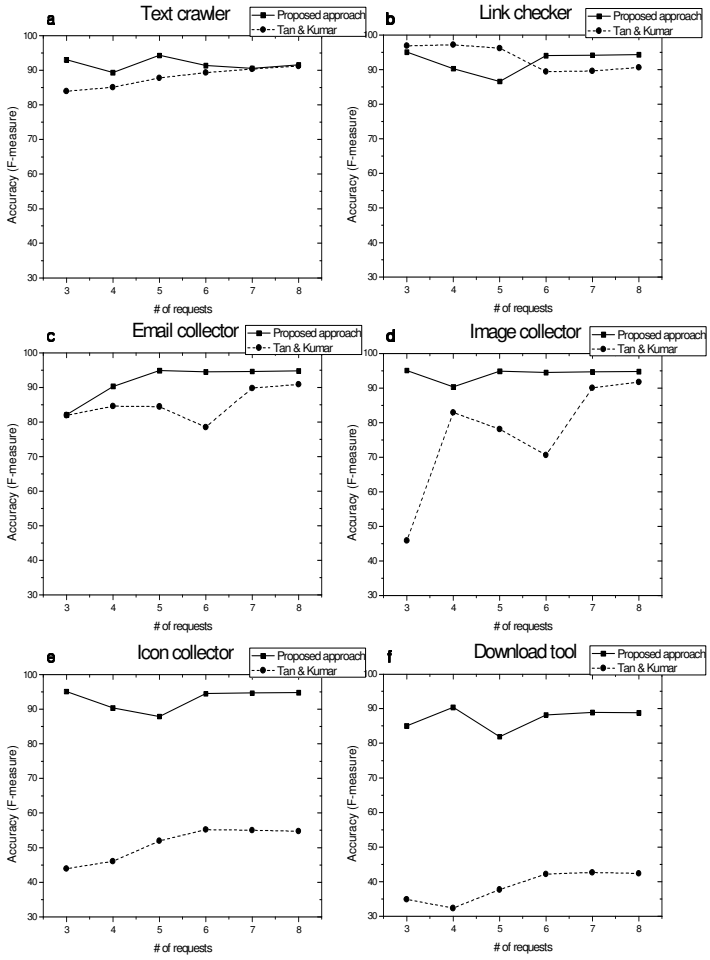

Fig. 2. F-measures grouped by web robot types on various session lengths

As already mentioned, the common features, such as switching factor, are one solution to imbalance problems. Although we have imbalanced samples, like other studies [2-6], we successively classified six types of web robots with similar performance. The decision tree using the proposed features achieved about 91% accuracy in a session composed of more than six requests. The download tool achieved relatively low performance (about 88% accuracy). This results in a greater switching factor in terms of packet size from client than other types of web robots, since download tools collect various types of files. This finally leads to low performance.

We were able to use 16 out of 26 features, in replicating an experiment reported in [2], due to differences in web log format. Our approach and that of Tan are both interested in text crawler, email collector, and link checker. Our approach in detecting these three types of web robots is slightly more accurate when sessions exceed six requests.

5 Conclusions and future works

This paper demonstrated that behavioral patterns are different between the sessions initiated by interactive users are fundamentally different from those triggered by web robots. Switching factors, characterizing the degree of uniformity in request patterns, have been proposed as effective features to detect web robots of various types. Proposed features works well on all types of major web robots found in the Microsoft web server log. Real-time detection of web robots seems feasible because proposed features are relatively simple to compute and small in numbers. Real world demonstration remains a major task for future research.

References

1. Doran, D., Gokhale, S.: Web robot detection techniques: overview and limitations. Data Mining and Knowledge Discovery 22(1),183-210 (2011)
2. Tan, P.N., Kumar, V.: Discovery of Web robot sessions based on their navigational patterns. Data Mining and Knowledge Discovery 6(1), 9–35 (2002)
3. Lourenco, A., Belo, O.: Catching web crawlers in the act. In: Proc. of the 6th international Conference on Web Engineering, ACM, pp. 265-272 (2006)
4. Bomhardt, C., Gaul, W., Schmidt-Thieme, L.: Web robot detection-preprocessing web logfiles for robot detection. New developments in classification and data analysis pp. 113-124 (2005)
5. Stassopoulou, A., Dikaiakos, M.D.: Web robot detection: A probabilistic reasoning approach. Computer Networks 53(3), 265–278 (2009)
6. Lu, W., Yu, S.: Web robot detection based on hidden markov model. In: Communications, Circuits and Systems Proc., 2006 International Conference on, IEEE, vol. 3, pp. 1806-1810 (2006)
7. Almeida, V., Menasc´e, D., Riedi, R., Peligrinelli, F., Fonseca, R., Meira Jr., W.: Analyzing Web robots and their impact on caching. In: Proc. of the Sixth Web Caching and Content Delivery Workshop (2001)
8. Huntington, P., Nicholas, D., Jamali, H.: Web robot detection in the scholarly information environment. Journal of Information Science 34(5), 726-741 (2008)
9. Duskin, O., Feitelson, D.G.: Distinguishing humans from robots in web search logs: preliminary results using query rates and intervals. In: Proc. of the Workshop on Web Search Click Data, pp. 15–19. ACM (2009)
10. Ye, S., Lu, G., Li, X.: Workload-aware web crawling and server workload detection. In: Proc. of Asia Pacific Advanced Network 2004
11. Lee, J., Cha, S., Lee, D., Lee, H.: Classification of web robots: An empirical study based on over one billion requests. Computers & Security 28(8), 795–802 (2009)
12. Quinlan, J.: C4. 5: programs for machine learning. Morgan kaufmann (1993)