Evaluation of Web Robot Discovery Techniques: A Benchmarking Study

Nick Geens, Johan Huysmans, and Jan Vanthienen

Department of Decision Sciences and Information Management,
Katholieke Universiteit Leuven, Naamsestraat 69,B-3000 Leuven, Belgium

Abstract. This paper describes part of a web usage mining study executed on log files obtained from a Belgian e-commerce company. From these log files, it can be observed that numerous web robots are active on the site. Most of these robots show a crawling behavior that is radically different from the browsing behavior of human visitors. Because the owners of the e-shop desire information about the paths that human visitors follow through the site, it is of crucial importance to remove these robotic visits from the log files.

Several existing methods for web robot discovery are evaluated and compared, none of them leading to satisfying results. Therefore, a new technique is developed that results in a successful and reliable identification of web robots.

1 Introduction

Web Usage Mining is defined as *the application of data mining techniques to discover usage patterns from web data* [1]. Usually, the web data that is being analyzed consists of log files that store information about the requests made to a particular web server over a certain time interval. In this paper, we discuss the analysis of the log files of a Belgian online shop. During this process the server logs were subjected to the subsequent steps of the typical web usage mining process [2,3,4,5,6]. This process consists of three parts: pre-processing of the data, pattern discovery and pattern analysis. In this paper, we will only describe a certain aspect of pre-processing, more specifically robot discovery. Robot discovery (also called robot identification or detection) is the search for robot sessions in a log file in order to exclude them from the analysis. A formal definition of web robots can be given as follows: "*Web robots are software programs or agents that traverse the hyperlink structure of the World Wide Web by retrieving a document and recursively retrieving all documents that are referenced*" [7]. By doing this, they are capable of automatically locating and retrieving information on the internet.

In the literature, several synonyms can be found for web robots, such as spiders, crawlers or web wanderers. The need for web robots was created by the soaring abundance of information on the internet. Nowadays, it has become an impossible challenge to find the required information on the web without the aid of a search engine. However, only few people realize that search engines succeed

P. Perner (Ed.): ICDM 2006, LNAI 4065, pp. 121–130, 2006.

in structuring this vast amount of information with the aid of web robots. Their crawling operations enable the search engines to store the visited web pages in indices. Besides these indexing robots, there are however plenty of other types of robots with less honorable intentions.

In this paper, we take a closer look at the different types of robots operating on the web and give an overview of several reasons why robot detection is worth the effort. Afterwards, we discuss some robot characteristics and currently applied methods to detect robot sessions. The remaining part of this paper will cover our practical research concerning the evaluation of the different robot detection methods and the development of a new method enabling a more reliable classification of robot sessions.

2 Motivations for Robot Discovery

There are many situations in which it is essential to separate robot sessions from visits of human users. First of all, when performing web usage mining on log files, it is indispensable to remove the robot sessions. The main purpose of this type of analysis is to extract useful information about the behavior of the human visitors of the site. Knowing that robots tend to have totally different browsing patterns compared to those of human visitors, the results of the analysis will be strongly biased because of robot presence.

Secondly, some e-commerce websites may contain information of high strategic value. Web robots can easily collect and aggregate this data, leading to precious business intelligence being exposed. To deal with this problem, preventive solutions will have to be developed in order to deny these robots access to the website.

Thirdly, a multitude of robots are employed by senders of spam to collect all email addresses that appear on web pages. Recognition of these malicious robots, can reduce the amount of spam received. Another possible approach followed by several sites is to show pages with non-existent email addresses when receiving a visit from these robots in order to pollute the databases of the spammers.

Another reason for robot detection is the excessive amount of bandwidth and server resources used by some robots. A great deal of web robots do not make use of these network resources in a responsible way, which can cause serious delays for other users.

Finally, some robots may be employed to perform fraudulent or illegal actions. When an advertising website is paid in relation to the amount of clicks it gets on the banners shown on the website, a robot can be designed to automatically click these advertisements in order to artificially inflate the number of banner clicks.

3 Different Types of Web Robots

As mentioned above, the most important task of robots is retrieving web pages for adding them to search engine indices. These **indexing robots** will be fed with a certain page (the ‘seed’) as a starting point. Due to the high level of connectivity between websites the robot will be able to take off on its journey through the internet.

Other robots are used to execute **link checking**. Since the internet is a rapidly changing and uncontrolled environment, web pages will be created, moved and deleted at all times. Nothing is more annoying for the visitors of a website to be confronted with links which lead them to the well known 404-page. In order to detect these broken links, robots can automatically check all the hyperlinks on a website and report dead links to the webmaster.

Thirdly, robots can be used to realise **offline browsing**. Users who want to be able to visit a web page at times when they are not online, can create an offline version of this website on their hard disks. All common browsers contain this kind of robot (e.g. MSIECrawler in Internet Explorer) which will download every page, image and other related file of a website. In some cases it might be interesting to duplicate websites or transfer them to another location. These mirrors will be created for websites with a large number of daily visitors in order to spread traffic over more web servers. For the sake of a faster response time, mirrors will sometimes be placed on different continents, enabling interaction with a web server residing nearby the client. Off course the consistency between different mirrors of one website must be guaranteed at all times: this task is perfectly executable by robots.

Another purpose for robots can be found in comparing prices of a given product on several e-commerce websites. The use of these so-called **shopbots** is a great asset for online customers, but will off course push the higher priced products out of the market. Therefore **pricebots** have been created for online merchants. These robots will dynamically adapt the prices of the offered products in function of the observed prices on other websites [8].

Finally, we may not ignore that robots might as well be employed in abusive practices. **Email harvesters** for example, travel through websites to collect email addresses which will be used for marketing and spam purposes.

4 Common Detection Methods Based on Log File Characteristics

We will now take a look at the typical characteristics of robot sessions in a log file. Some current methods for robot discovery based upon these characteristics will be discussed as well as the inherent flaws of these techniques. Related research on this topic was conducted in [9] and [10].

We start with the most frequently used method, which rests on the Robot Exclusion Standard of Koster [11]. This protocol proposes the use of a file named ‘robots.txt’ to indicate which parts of a website are restricted for robot visits. According to this standard, robots should always consult this file before indexing a website. Since human visitors will, during normal use, never end up at this file, a robots.txt request in the log file is strong evidence for the presence of a robot session. If all robots obeyed the Robot Exclusion Standard, this method would yield a perfect detection and there would be no need for robot discovery research. However, since these rules can not officially be imposed, there is no guarantee that all robots employ this standard.

Another typical robot feature can be found in the user agent field of the log file. Normally, information about the browser type used by the visitor can be found in this field. In the case of robots however, Eichmann [12] has postulated in his Directives for ethical web agents that they should make use of this field to declare their identity. When going through a log file, one will in fact notice that the user agent field in some sessions contains a robot name followed by some additional information. For instance, when Google has indexed a website, the log file will show the visit of their bot as follows: "Googlebot/2.1 (+http://www.googlebot.com/bot.html)". As a consequence, checking these user agent fields for names which are related to robots (e.g. all names containing 'bot', 'spider', 'crawler' etc.) has become a widely applied method for robot discovery. Eichmann's principles however are predestined to the same fate as the Robot Exclusion Standard, because these suggestions are also not enforceable to robot designers. Some robots go even further and try to conceal their identity by mentioning user agent information belonging to regular browsers.

Because the robots.txt request and user agent information are features which can be determined by robot designers themselves, there is clearly a need for methods relying on objective robot identification. From this point of view, the approach has been suggested to create a list containing IP addresses of all known robots and comparing these addresses to the ones in the log file. The soaring amount of robots operating on the web however, makes the maintenance of such a list a virtually impossible task. Moreover, the presence of proxy servers and providers with IP address pools has made it impossible to determine exactly which users belong to which IP addresses.

As stated above, robots will often attempt to remain undetected. This is why some existing methods try to discover robots based on their typical crawling behavior rather than using features referring to their identity. Detecting this robot behavior can first of all be achieved by considering the method (e.g. GET, POST, HEAD etc.) used to perform the HTTP requests. Browsers of human users will always request web pages with the GET method in order to receive the complete HTML page from the server. The task of some robots (e.g. link checking) on the other hand, can in several cases be executed by applying the HEAD method, causing only the header of the HTTP message to be transferred across the network. Another approach takes the referrer field of the logfile into account. This referrer field shows the web page that contains the link followed by the client in order to reach the requested page. In some specific situations however, this referrer field will not be registered, which is denoted by a hyphen. These unassigned referrers will occur when a visitor has manually typed the URL of a page in his browser or when he has reached a given page through his bookmarks. Robots in particular do often not assign a value to the referrer field, leading to all requests of their session having an unassigned referrer.

Three other methods based on the browsing behavior of web robots are related to the time pattern of subsequent requests. First of all, the ethical robot directives state that bandwidth should not be overconsumed at the expense of human users. Therefore, robots would have to operate as much as possible during the night. Secondly, from the same point of view, robots should insert a waiting period between subsequent requests instead of firing requests at the server every other second. If robots employ such a fixed request delay, this will result in a zero standard deviation in the times between subsequent requests. On the other hand, some robots are not considerate of human users and overload a server with requests in a short period of time. That is why a very low average time between subsequent requests is also a strong indication for robot sessions. As we will discover in our practical research however, none of these time-related techniques succeeds in efficiently distinguishing robots from human visitors.

Finally, we mention a last characterizing aspect of robot sessions which has not yet been included in standard robot discovery techniques. Remember that using the HEAD method was justified since for some robot purposes it appeared to be unnecessary to request complete web pages. Furthermore, most robots are also not interested in the images embedded in web pages as they are unable to extract useful information from these images. As a consequence, we can distinguish these robots from human visitors, whose browser will automatically depict all images belonging to a requested web page. The absence of image requests however, is not a guarantee that we are dealing with a web robot, since some visitors may have adjusted their browser settings in such a way that images are not shown.

5 Practical Evaluation of Currently Applied Methods

In order to assess the described methods, we have performed a practical study on the log files of a Belgian e-commerce website. Before discussing the results, we introduce two criteria on which this evaluation will be based. We define recall and precision as follows:

$$\textbf{Recall} = \frac{\text{number of correctly identified robot sessions}}{\text{total number of actual robot sessions}} \tag{1}$$

$$\textbf{Precision} = \frac{\text{number of correctly identified robot sessions}}{\text{total number of predicted robot sessions}} \tag{2}$$

In other words, recall yields the percentage of robot sessions that were discovered using a particular method, whereas precision describes the accurateness in terms of the proportion of correct predictions. It is clear that any method, in order to be useful, has to obtain a sufficient score on both metrics. There is no advantage in being able to detect all robot sessions if at the same time half of the predictions is incorrect.

Of course, there is one condition linked to these criteria: we must know exactly which sessions in the log file were created by robots. Therefore, we have manually

Table 1. Evaluation of commonly applied methods

	Correct	Wrong	Recall(%)	Precision
Manual Research	241	0	100	100
Robots.txt	41	0	17.01	100
IP address list	167	0	69.29	99.4
Robotic User Agent	64	0	26.56	100
HEAD method	78	0	32.37	100
Unassigned Referrer(1-100)	232	212	96.27	52.25
No Image Requests	237	77	98.34	75.48
Night	59	58	24.48	50.43
Standard Deviation (3s)	6	0	2.49	100
Average Time (1s)	6	2	2.49	75

checked a period of 5 days, resulting in 241 robot sessions out of a total 8001 registered sessions. In order to be sure about the origin of the visitors, we checked each session on the simultaneous presence of several robot characteristics and made use of DNS reverse look-up when this examination could not give a decisive answer.

Table 1 summarizes the outcome of each of the 9 considered methods. Before evaluating these methods, we quickly go through some implementation details. The list of known robots was based upon the overview available on www.robotstxt.org [13], while the third method scanned the user agents for the following words: bot, crawl, search, seek, archive, scan, link and spider. According to the HEAD method-technique, sessions are considered to be robotic as soon as they contain one occurrence of the HEAD method. Unassigned referrers (1-100) means that we detect sessions satisfying two conditions: the minimum number of requests in the session is 1 and all of the requests (100%) must contain an unassigned referrer. The night feature selects all sessions falling between 00.00 am and 07.00 am. Furthermore, we will consider all sessions as robotic if the period of time between subsequent requests has a standard deviation of less then 3 seconds or an average of less than 1 second. All of these parameter values were deduced from a separately conducted experiment which we will not treat in-depth here.

Considering the results for recall and precision, we can distinguish three groups of methods.

- The first group of 4 methods (robots.txt, IP address list, robotic user agents and HEAD method) are those with a perfect precision, but a rather low recall. A precision of 100% means that during the examined period, these 4 characteristics could only be found in robot sessions, making them very reliable techniques to discover robots. However, regarding the poor recall values, this reliability is not worth a great deal since these methods only detect about 20 to 30% of the robot sessions. Only by using the IP address list a considerably higher recall of 70% could be obtained, but this result must be nuanced knowing that the log files we examined were dated from

3 years before the applied IP address list. This way, all robots operating at the time of the log files, have probably been discovered and registered on the list by now.

- A second coherent group are the methods based on unassigned referrers and the absence of image requests. These techniques manage to discover almost all robot sessions, but also lead to a great amount of false positives.
- Thirdly, we notice that the techniques based on time-related features do not score well at all. A lot of robots seem to operate during the daytime and human visitors tend to execute nocturnal sessions as well. This is not surprising taking the high level of international web traffic into account. The methods ‘standard deviation’ and ‘average time’ are also incapable of detecting a sufficient percentage of robot sessions. The main reason for this failure can be found in the presence of the large amount of robot sessions existing out of only one request. Of course this type of sessions can not be detected by methods needing at least two requests to calculate standard deviation and average values. More robots could be discovered by applying higher maximum values for the parameters, but we found that this results in a plummeting reliability, making these methods completely useless.

It is obvious that applying any of the considered methods will result in a large part of the robot sessions remaining undetected or in misclassifying a substantial amount of human users as robots. Notice that the proposed Robot Exclusion Standard results in a very poor detection of only 17% of the robot sessions. The ethical guidelines on the other hand, are also ignored by most of the robots, regarding the low recall values for techniques based on requests during night time and on user agent information. All together, we can conclude that the currently existing techniques are inadequate to execute robot discovery in an accurate fashion.

6 Proposal for a New Robot Discovery Technique

If we want to develop a new method for robot detection, it is essential that it yields a high level of recall and precision at the same time. Thus, our goal is to create one single technique combining the positive effects of both method groups but also excluding their deficiencies.

First of all, the 4 features resulting in a perfect precision will definitely have to be part of the composed method, since they do not damage the final solution in terms of reliability. We combine them into one new technique, which we will be referring to as the ‘high precision’ method further on. The characteristics are logically combined in such a way that as soon as a session complies with one of the conditions, the session will be labelled as robotic. We know in advance that this technique will yield a 100% precision and the recall value will be at least 69.29% (this is the highest recall of the 4 selected features). When calculating the results, we notice that this method detects 73% of the robots sessions, an increase of only 4% compared to the IP address list method.

Table 2. Evaluation of combined robot discovery methods

	Correct	Wrong	Recall(%)	Precision(%)
Manual Research	241	0	100	100
High Precision (H.P.)	176	0	73.03	100
H.P. or No images	240	77	99.59	75.51
H.P. or Unass. Referrer	236	212	97.93	52.68
H.P. or Unass. Referrer or No Images	241	260	100	48.10
H.P. or (Unass. Referrer and No Images)	235	28	97.51	89.35

In order to achieve a better recognition of robot sessions, we will be obligated to select one of the methods with higher recall values. By adding these less reliable characteristics to our composed technique, we expect our global solution to be penalized in terms of precision. Indeed, the high precision method in combination with the image absence feature delivers an almost perfect detection of 99%, but the precision tumbles to 75%. The same effect can be observed when we apply high precision together with the unassigned referrers characteristic. Recall is improved to almost 98%, while precision drops to 52%, which is even lower than in the previous case. It is remarkable in those two results that precision falls back to a level which more or less corresponds to the individual precision of the added methods. The performance of a combined method seems to be completely determined by the strength of its weakest link. For completeness reasons, we also mention the results of combining all methods in one composed technique, however these results are -as logically expected- even worse than the ones above. If a session is classified as a robot session when it complies with one of the 6 robot features, a recall of 100% and a precision of 48% is obtained.

To upgrade our composed method we will have to look for ways to strengthen the weakest link. Therefore we examine the types of false positives occurring in the individual implementations of the two unreliable methods. On one hand, it can be noticed that 'No images' misjudges sessions of human clients with particular browser settings causing embedded images not to be requested. Incorrect evaluations of 'Unassigned referrer' on the other hand, appear to be short sessions of visitors entering the site by manually typing the site's URL and immediately leaving afterwards. These two types of misclassifications are not correlated, so we may assume that a given session will only coincidentally be misclassified by both methods at the same time. In other words, the cross-section of these two false positives sets will be more or less empty. Considering this observations, it makes perfect sense that by combining methods in the way we did above, the global result of one technique was determined by the weakest link. Remember that the composed methods were based on a logical OR operator: as soon as one of the robot conditions was fulfilled, the session was deemed to be robotic.

In order to exclude all false positives situated outside this cross-section, we now combine the two unreliable methods by means of a logical AND operator before adding them to the High precision technique. The full definition of this technique then becomes:

"Robots.txt OR IP address list OR Robotic user agent OR HEAD method OR (Unassigned referrer AND No images)"

This composed technique should offer the recall power of the methods 'Unassigned referrers' and 'No images' in combination with an acceptable precision. In fact, the practical results showed us a recall value of 97.51% while still guaranteeing a reliability of 89.35%. In comparison to the other composed techniques, this is a stunning improvement, as can be seen in the overview given in Table 2.

7 Conclusion

In this paper, robot discovery as a part of web usage mining was treated and the currently applied techniques for robot detection were discussed. The main part of this paper dealt with the practical evaluation of these methods, which learned us that none of them managed to accurately classify robot sessions. Consequently, we discussed some possible composed techniques in order to obtain better results and proposed the use of a method which succeeds to detect almost every robot session with a reliability reaching up to 90%.

References

1. Srivastava, J., Cooley, R., Deshpande, M., Tan, P.N.: Web usage mining: Discovery and applications of usage patterns from web data. SIGKDD Explorations **1**(2) (2000) 12–23
2. Cooley, R.: Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data. PhD thesis, University of Minnesota (2000)
3. Huysmans, J., Baesens, B., Vanthienen, J.: Web usage mining: a practical study. In: Twelfth Conference on Knowledge Acquisition and Management (KAM 2004). (2004)
4. Perner, P., Fiss, G.: Intelligent e-marketing with web mining, personalization, and user-adapted interfaces. In: Industrial Conference on Data Mining (ICDM02), London, UK, Springer-Verlag (2002) 37–52
5. Blanc, E., Giudici, P.: Sequence rules for web clickstream analysis. In: Industrial Conference on Data Mining (ICDM02), London, UK, Springer-Verlag (2002) 1–14
6. Huysmans, J., Baesens, B., Mues, C., Vanthienen, J.: Web usage mining with time constrained association rules. In: Proceedings of the Sixth International Conference on Enterprise Information Systems (ICEIS 2004), Porto, Portugal (2004) 343–348
7. Heinonen, O., Hatonen, K., Klemettinen, K.: WWW robots and search engines (1996) Seminar on Mobile Code, Report TKO-C79, Helsinki University of Technology, Department of Computer Science.
8. Greenwald, A.R., Kephart, J.O.: Shopbots and pricebots. In: Agent Mediated Electronic Commerce (IJCAI Workshop). (1999) 1–23
9. Almeida, V., Menasce, D.A., Riedi, R.H., Peligrinelli, F., Fonseca, R.C., Jr., W.M.: Analyzing web robots and their impact on caching. In: 6th Web Caching and Content Delivery Workshop. (2001) 299–310

10. Tan, P., Kumar, V.: Discovery of web robot sessions based on their navigational patterns. Data Mining and Knowledge Discovery **6** (2002) 9–35
11. Koster, M.: The robot exclusion standard (http://www.robotstxt.org/wc/norobots.html) (1994)
12. Eichmann, D.: Ethical Web agents. Computer Networks and ISDN Systems **28**(1–2) (1995) 127–136
13. Koster, M.: The web robots database (http://www.robotstxt.org/wc/active.html) (2004)