Where is the user? Filtering Bots from the Edurep Query Logs

Wim Muskee

Kennisnet Foundation
Paletsingel 32
2718 NT Zoetermeer, NL
w.muskee@kennisnet.nl

Abstract. Edurep indexes learning object metadata from several repositories, offering a webservice interface on which portals can build their own search implementation.
At Edurep query log level, no obvious distinction can be made between human users and webcrawlers visiting these portal sites. This makes it impossible to gather any meaningful data on user search behaviour.
Four query types, distinguished from the six largest portals' websites were related to one month of query logs. For two query types a distinction between human and automatic generated traffic could be found. However, these results can only be used to advise connected portals on their interface implementations. More research is needed to actually perform any reliable filtering.

Keywords: webservice, crawler detection, log analysis

1 Introduction

Edurep is a Dutch learning object search engine, indexing harvested learning object metadata from more than 50 different repositories. Search portal developers can interface with the search engine using the Edurep webservice, available through the SRU/SRW protocol (Figure 1).
Although operational for some years [11], the operators gained access to the search query logs only recently (december 2009). Through analysis of these logs and webserver logs of one portal, the operators discovered that a significant amount of queries came from various search engine bots[1]. Among several harm-

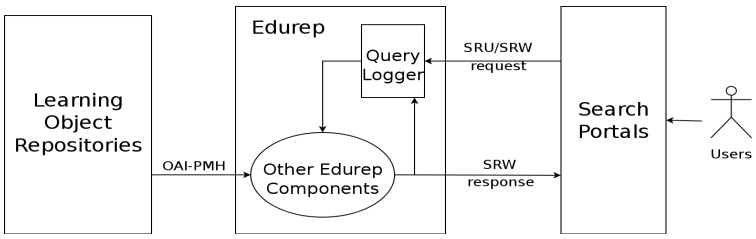

Fig. 1. A simplified diagram of Edurep in its context.

ful aspects, Edurep is affected by two in particular. First, and obviously, webcrawlers generate extra traffic, possibly limiting performance for human users. Secondly, webcrawlers generate automated traffic, making it harder for the operators to infer meaningful human interaction results from the Edurep query logs.

Most of these search engine bots can be identified at search portal level based on their HTTP request User-agent string or IP adress [12,9]. However, this information is no longer available when the request reaches Edurep.
This problem is not typical for Edurep, but applies to any webservice which allows connections from a third-party search interface. Examples of these in the learning object context include the LRE [3], MACE [14] and the Spider project [4], all of them available through the SQI protocol [7].

With Edurep as context, this paper aims to explore methods to make a distinction between automated and human queries in webservice query logs. To this end, four query types were distinguished from several search portal web interfaces. The SRU representations for each query were used to filter the logs for a specific query type and analyze it more closely. The paper ends with a discussion of the results.

[1] A type of webcrawler; a program which gathers information from the internet by recursively following found hyperlinks.

2 Modeling Automated Queries

Because webcrawlers only follow hyperlinks, automated searches are caused by the presence of hyperlinks which cause an Edurep search query. An analysis of the portals' search interfaces is necessary to combine hyperlinks with logged SRU queries.

2.1 Portal Search Interfaces

Looking at the search interfaces of the six largest portals (consisting of 97% of query total), four types of hyperlinks were distinguished.

- *search links:* Issuing a search to retrieve a first page resultset.
- *pagination links:* Issuing a search to retrieve another resultset page.
- *result links:* Issuing a search to retrieve a specific record.
- *facet links:* Issuing a search to retrieve the amount of records for that facet.

Typically, the portals retrieved either 5 or 10 results after a search query. The number of navigation links ranged from 5 to 20, always including a next and/or previous link and sometimes including links to the first and/or last page. A few included result and facet links.
Only one portal (C) performed a search on page arrival. The resulting page included all link types. All the portals' queries were represented as a url in the browser navigation bar, meaning they can be pasted easily on other webpages for others to click on, including bots. When searching for the portals' url query prefixes on Google, indeed some results were found. Also corresponding queries were discovered in the query logs.

2.2 SRU/SRW

Edurep can be queried using the *searchRetrieve* operation of the SRU/SRW protocol [13]. Among several supported request parameters [10], the *startRecord* parameter determines which record of the resultset is displayed first. When omitted, it defaults to 1. The *maximumRecords* parameter sets the number of records each resultset contains. Edurep's default is 10.
A search query typically has no *startRecord* value at all or a value of 1. Also, to present a reasonable amount of results, the *maximumRecords* value is set to 5 or higher, or left out to return 10. Pagination queries have a *startRecord* value higher than 1.
In a result query, the *startRecord* value is omitted or 1. Since a result of 1 is expected, the value for *maximumRecords* does not need to be 1. However, because a specific record is requested, part of the *query* value is characteristic. In Edurep, a specific record can be requested by filtering on lom.general.identifier or lom.general.catalogentry, the LOM identifier, or meta.upload.id, Edurep's internal unique identifier.
Facet queries can be performed inside a search query by adding Edurep's *x-term-drilldown* parameter to the SRU query. In addition to the search results,

a count drilldown for each facet of the requested field is retrieved. Because this function is not supported for all LOM fields, separate facet queries can also be executed. These have a *startRecord* value of 1 or none at all. Also, the value for *maximumRecords* is 0 or 1 [2].

3 Dataset

The logs of January 2010 were used as dataset and the analysis is done in R [5]. Each log entry consisted of the portal's ip adress, the timestamp when a search query entered the system (UTC), the size of the response data in kilobytes, the processing time in seconds, the entrypoint of a query on the server indicating the used protocol (SRU or SRW), and the SRU search query.
Five variables from each query were used. The IP adress, *startRecord* and *maximumRecords* values were used unprocessed. The *query* argument was used as a whole, assuming each portal constructed their queries in the same way and query uniqueness was not compared across portals. An identifier boolean was set to 1 if a result link was detected.

4 Results

Concerning *search queries*, the distinction between human and automatic induced queries can be made based on the occurence of the queries. Automatic induced queries will appear more often in relation to human generated ones. While Portal C's startup page query appeared more than 6 times than any of its other queries, a good threshold could not be determined.
Assuming most users will never click past the second page of search results [1], *facet queries* with a *startRecord* value over 200 will probably be auto-generated (PAG1). A more elegant method for determing automatic facet queries is to scan the logs for pagination ranges. A range was crudely defined as a set of SRU queries (min. 10) with equal *query* values, a *startRecord* difference of *maximumRecords* and a maximum *startRecord* value higher than 200 (PAG2).
Based on occurence of *result queries*, no clear evidence for automatic querying was found in the logs. This was attributed to the dynamic nature of Edurep's content, with changing resultsets, different results will be queried.
After plotting the unique *facet queries* of Portal C (Figure 2), the small layer of queries below the top coincided with the facet queries executed on entering the search page. Observing that 10 of the 12 sub-top queries were executed about 2330 times, it was assumed they were caused by automatic querying. From the queries of these types, that amount could be subtracted, leaving their human induced occurences (FACET). Following from this assumption, at least the same amount of automatic hits were generated by Portal C's startup search query, and could thus also be subtracted.

[2] Technically, by setting this value to 0, the same total can be retrieved, but since the usage of 1 had been observed, it was included

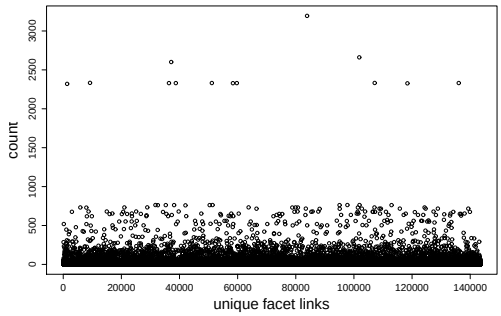

Fig. 2. Unique facet link queries plotted against occurences.

	total	PAG1	PAG2	FACET
Portal A	41690	-15237	-13355	
Portal B	126340	-105026	-89710	
Portal C	1293902	-15255	-15654	-30290
Portal D	48841	-47	-62	
Portal E	232341	-1778	-1815	
Portal F	82527	-406	-205	
total	1825641	-137749	-120801	-30290

Table 1. Subtractions of filter method implementations..

The subtractions from each filtering method are displayed next to to each portal's total amount of queries in table 1.

5 Discussion

Considerable automatic induced querying was observed. In terms of bandwidth the found ranges from *PAG2* alone caused 13,3 Gb of traffic, $26,5\%$ of the total A-F amount. Concerning the amount of queries, *PAG2* and *FACET* accounted for $8,4\%$ of the total A-F amount of queries.
However, assumptions were made and the used filter methods are still rudimentary and incomplete. In using *PAG2* for instance, tails or heads of the ranges may lie outside the used dataset. Also, the dataset probably contains heads or tails of ranges from other months. This is even more true when considering the pagination queries don't need to appear on the timeline in the same order as they appear on the page [2]. Secondly, first- and lastpage pagination queries were not considered in *PAG2*.

The immediate findings of this study make it possible to tailor our advise for portals. One aspect of this is related to blocking crawlers at the portal by implementing the Robots Exclusion Standard [6]. Use of this standard could also be enforced through Edurep's user level agreement. As an unintended side effect, automated usage amplified some examples of inefficient quering on Edurep. Another aspect of the advise should include information on how to interface with Edurep better.
Use of various scripts to parse and filter the log files proved very useful during the conduct of this study. Automating the used scripts will allow the administrators to detect undesirable behaviour in an earlier stage and act on it sooner, leaving Edurep free to be used by actual users.

Future research should improve on several aspects. First of all, more months of logging need to be used to combine and compare with current results. Secondly, the SRU *query* values need to be parsed fully to allow more accurate filtering options and to compare queries across portals. Last is the usage of the portal website. Parameters like the size and format of the pagination links, and the types of search, result or facet links on the page could prove useful in implementing better automatic detection methods.

A more long term product change would be to also request the end user's orginal User agent string in the query to Edurep. Also requesting the original IP adress could lead to privacy concerns. Since lots of crawler User agent strings are publicly available [8], this information could greatly enhance our filtering efforts.
An new Edurep component could be introduced, making it possible to block requests before they are processed by the system. However, at this point it is unclear if such an extra check on all requests outweighs the benefits of not having to process the blocked requests. For now, such a filtering component will have to implemented before the logs are processed by our business level reporting tool.

While the ideas in this paper could be used in similar architectures, the actual scripts cannot because they are made for SRU and Edurep's query log format. With more standardization in repository query languages (like SQI), corresponding logging standards can be thought of, making sure developed analysis tools benefit many and query logs can be shared easily.
Filtering automatic queries is after all needed to look more closely at the human ones. The focus of interest is teacher search behaviour, not only on Edurep but beyond our borders.

References

1. Aula, A.: Studying user strategies and characteristics for developing web search interfaces. Dissertations in Interactive Technology 3 (December 2005)
2. Dikaiakos, M.D., Stassopoulou, A., Papageorgiou, L.: An investigation of web crawler behavior: characterization and metrics. Computer Communications 28(8), 880–897 (2005)

3. Massart, D.: Towards a pan-european learning resource exchange infrastructure. Lecture Notes in Computer Science 5831/2009, 121–132 (2009)
4. Paulsson, F.: Connecting learning object repositories: Strategies, technologies and issues. Internet and Web Applications and Services, International Conference on 0, 583–589 (2009)
5. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2009)
6. robotstxt.org: The web robots page. Retrieved August, 3 2010, from http://www.robotstxt.org. (2007)
7. Simon, B., Massart, D., van Assche, F., Ternier, S., Duval, E., Brantner, S., Olmedilla, D., Miklós, Z.: A simple query interface for interoperable learning repositories. In: Proceedings of the 1st Workshop On Interoperability of Web-Based Educational Systems. pp. 11–18 (2005), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.67.7745
8. Staeding, A.: List of user-agents (spiders, robots, browser). Stichting Kennisnet. Edurep wiki. Retrieved August 5, 2010, from http://www.user-agents.org
9. Stassopoulou, A., Dikaiakos, M.: Web robot detection: A probabilistic reasoning approach. Computer Networks 53(3), 265–278 (February 2009)
10. Stichting Kennisnet: Edurep wiki. Retrieved June, 3 2010, from http://edurep.wiki.kennisnet.nl
11. Stichting Kennisnet ICT op School: De educatieve contentketen: leertechnologische afspraken voor de toekomst. Retrieved May, 2 2007, from http://contentketen.kennisnet.nl/attachments/990312/De_Educatieve_contentketen_-_Leertechnologische_afspraken_voor_de_toekomst.pdf (December 2006)
12. Tan, P.N., Kumar, V.: Discovery of web robot sessions based on their navigational patterns. Data Mining and Knowledge Discovery 6(1), 9–35 (January 2002)
13. The Library of Congress: Sru: Search/retrieval via url. Stichting Kennisnet. Edurep wiki. Retrieved August 5, 2010, from http://www.loc.gov/standards/sru/., http://www.loc.gov/standards/sru/
14. Wolpers, M., Memmel, M., Klerkx, J., Parra, G., Vandeputte, B., Duval, E., Schirru, R., Niemann, K.: Bridging repositories to form the mace experience. New Review of Information Networking 14(2), 102–116 (2008)