Compressing Weblogs for Efficient Discovery of Internet User Activities

Attila Babos, Sándor Juhász

Department of Automation and Applied Informatics
Budapest University of Technology and Economy
{attila.babos, sandor.juhasz}@aut.bme.hu

Abstract: It is very important for Internet content providers to keep track of the number visitors of their sites. Web auditing companies record downloads in enormous weblogs and use complex techniques to estimate the number of visitors based on this information. In this paper an efficient weblog compressing method is introduced, thus the complex data processing methods can be executed faster and more efficiently.

Keywords: Data Mining, Web Auditing, Weblog, Internet User, Cookie, Cookie-Chain, Dictionary Based Compression

1 Introduction

As the Internet became popular and the amount of web sites visitors increased dramatically, thus the online media became an important part of the advertisement market. Plenty of web audit companies [1] [2] [3] [4] measure the number of visitors by web sites in an automated way. The advertisers are not only interested in the amount of downloads of a given site, but also, in fact, they would like to estimate how many real persons can be reached by their advertisements. The amount of real persons cannot be derived directly from the gathered log, as such information is not available at user downloads. To each download only a so called Internet user can be assigned. An Internet user is identified by the computer, the username, and the browser currently used by the real person when browsing the Internet. [5] This triple can be tracked during the web auditing process. This means if a real person uses multiple computers, accounts, or browsers, then he or she will be considered as more Internet users. Counting the Internet users in the log files serve as base for calculating the number of real persons with complex business logic.

The rest of the paper is organized as follows. First in Section 2 the detailed structure of a weblog entry is described then in Section 3 a compressing method is introduced that allows creating a smaller data structure, which can replace the original log files in the discovering process of the Internet users. After that in

Section 4 the efficiency of the compressing method is discussed along with the conclusions and future plans.

2 The Structure of a Weblog

The weblogs of auditing companies [1], [2], [3], [4], contain all information which is needed for identification of Internet users. The most common identification method is the use of third party (C3) cookies [6], along with the use of first party (C1) cookie [7]. The main difference between the two types of cookies is that while C1 cookies are set by the page browsed currently, C3 cookies are placed by foreign, third party site referred from the currently browsed site. Since web pages often contain advertisements from foreign sites [3], they can also set C3 cookies on the clients, but client-side security software might remove them periodically. The cookie-based user identification methods are able to identify and trace Internet users; however they are unable to associate Internet users with real persons. An additional identification technique is needed to solve this problem. Websites having registration database (e.g. e-mail providers) identify their users with special unique login identifiers (noted as MID-s). In order to protect the privacy of real persons, even MID-s do not reveal personal information but they can be used to separate real persons from each other. [1], [3], [6]

Weblogs may contain not only these fields but also some others (e. g. IP address, time, URL, browser information) that allows making different kind of statistics about the Internet users and web site visitors. Currently 40-50 million entries are collected per day, which means a daily increment is about 16-20 GB. [1] Because of this enormously increasing database, a data compression is necessary especially if multiple reading is required for further processing. During the compression we are creating from the original data structure a smaller one that contains only the most important elements (Figure 1).

Original Structure	
Field Name	Size [Byte]
C3	22
C1	26
MID	40
IP Address	4
Timestamp	4
UCode	16
URL	100
Browser	180
Summary	392

Compressed Structure	
Field Name	Size [Byte]
C3	8
C1	16
MID	8

Figure 1
Structure of the original and the compressed weblog

3 Compressing

The complete data processing is a very complex task. All the weblogs can't be stored on the processing computer due to its huge size so at first, before processing a weblog has to be downloaded. It is about 40-50 GB per day, stored on the server compressed by a general compressing method. Downloading and uncompressing it takes approximately two minutes according to our measurements, but it naturally depends on the current download rate. The size of a record in the original weblog is 392 byte. The basis of the Internet user identification is its first three fields. The C1 field can be compressed easily lossless to 8 bytes (2 integers with computer representation) because of its special structure [1]. The C3 and the MID identifiers contain also too much redundancy, but they cannot be mapped easily by a simple function to one or two integers, so they are coded using hashtables. There are surely less MID values than 2^{32}, so they have a code of 4 byte length (one integer). There are much more third party cookies, but according to our current measurements it is not more than 2^{32}, so they are also coded to 4 bytes.

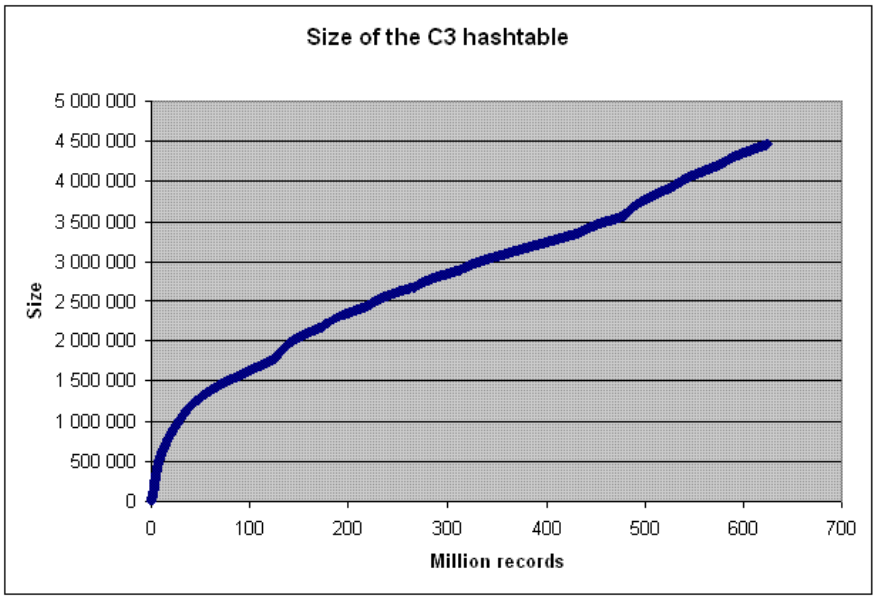

Figure 2
Size of the C3 hashtable

The processing time of the compressing step depends on the size of the hashtables (coding the C1 takes constant time). Figures 2 and 4 show the size of the C3 and MID hashtables after processing a certain amount records, while Figures 3 and 5 shows the growth of them, how many new element were added to the hashtables during the process of the last 100.000 records. Two weeks were processed, during

this time nearly 625 million entries were generated into the weblog. After finding the currently used third party cookies in the first days, the increment is fairly constant. In case of the MID-s, the saturation is spectacular although there is a step just before the end of the two week interval, caused by an agreement with a large portal that provides MID-s. In spite of this sudden increment the MID hashtable size is still much lower than the size of the C3 hashtable and it also grows slower, the growth of the C3 hashtable is constantly above 500 for each 100.000 records which was exceeded only for a short time by the MID hashtable. In order to maintaining and limiting the size of the C3 hashtable, a novel algorithm will be introduced which deletes the unused third party cookie identifiers, supposing that the cookie has already been removed on the client side computer.

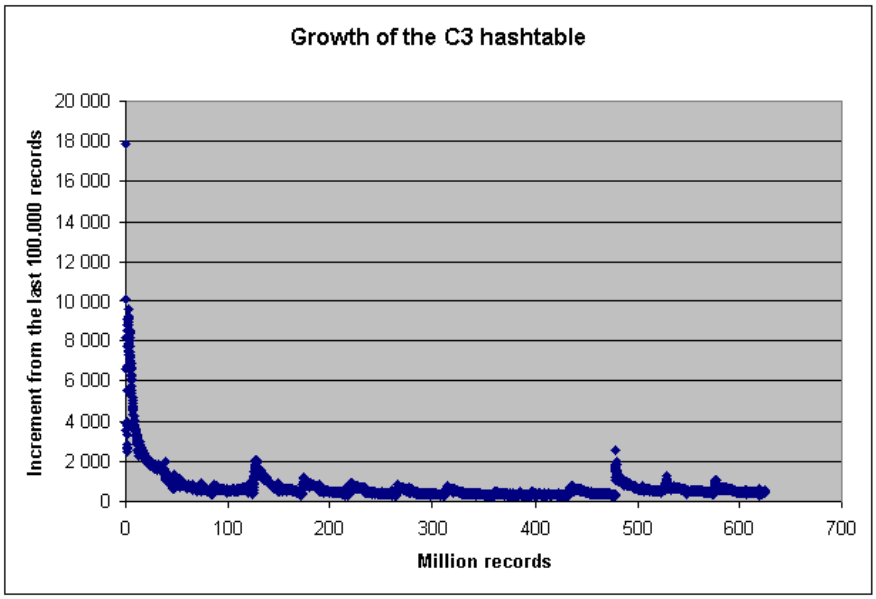

Figure 3
Growth of the C3 hashtable

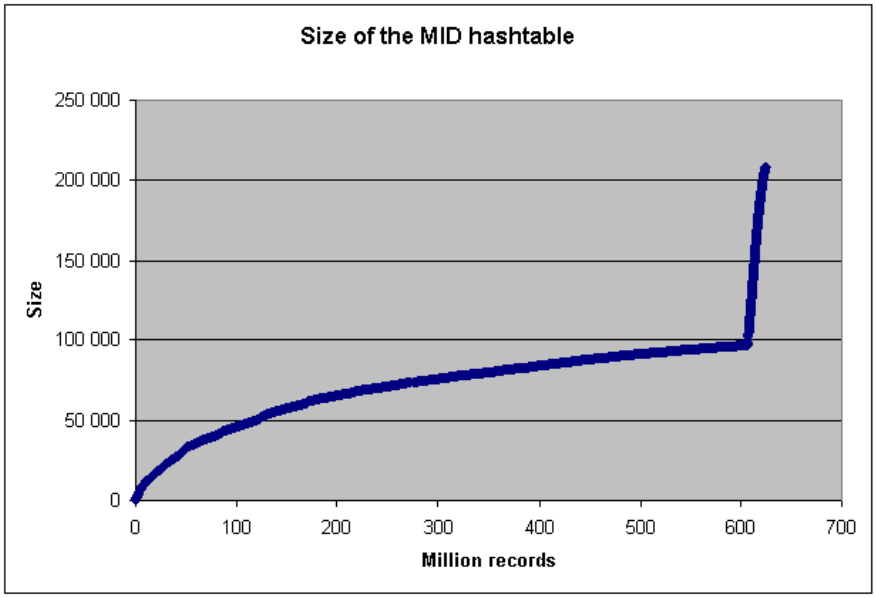

Figure 4
Size of the MID hashtable

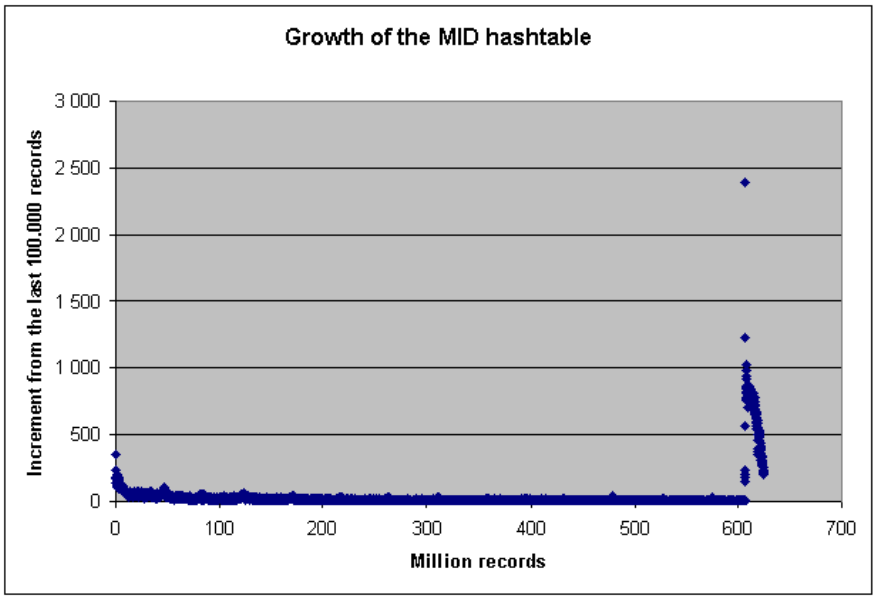

Figure 5
Growth of the MID hashtable

4 Experimental Results

The compressing algorithm was executed on a 13 day log of 123 files, containing 575 million records in total. Figure 6 shows the total process time while Figure 7 shows the process time of the last 100.000 records. The time needed for processing this amount of records slightly increases. The processing time of the last day (48.4 million records) was 3112 seconds without the download and uncompress phases, but reading the file on an asynchronous way. Skipping the main processing phase, just reading the file took 53 seconds, so the complete compressing method is rather CPU intensive than I/O. It allows us to do the download and the uncompress phase in another thread, parallel to our compressing phase. The further processing steps leading to the Internet users use only the compressed file; therefore their algorithm can run parallel on a separate computer.

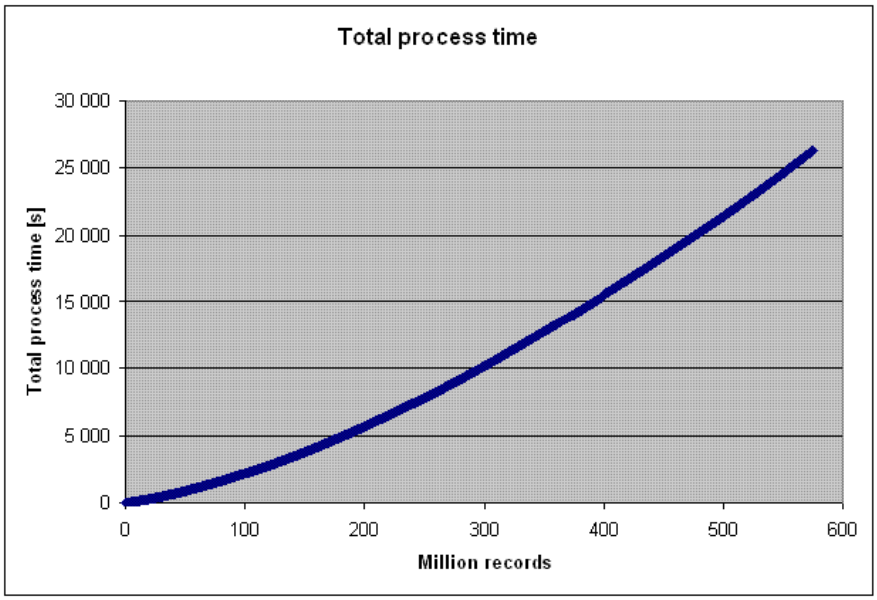

Figure 6
Total process time

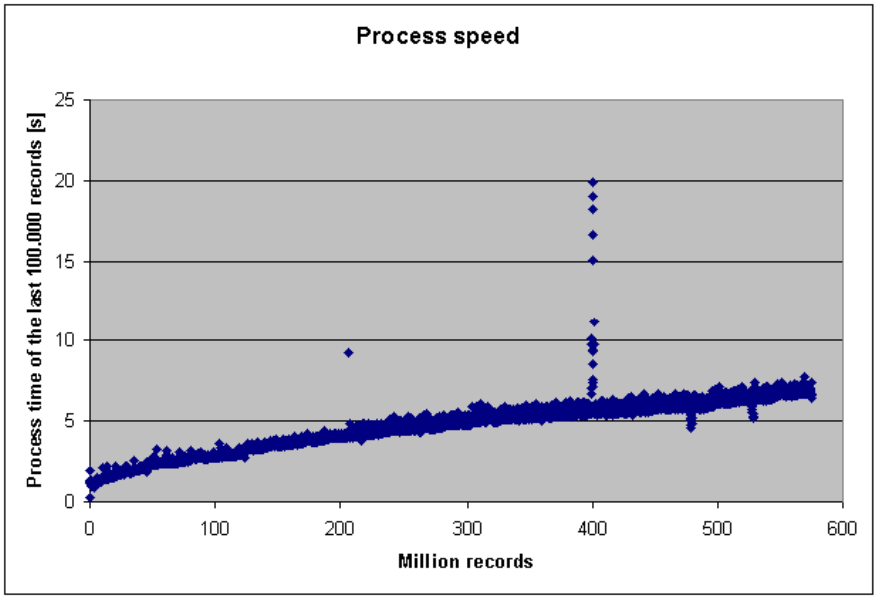

Figure 7
Process time of the last 100.000 records

The measurements were made on the following computer:

OS Name	Microsoft Windows XP Professional
Version	5.1.2600 Service Pack 2 Build 2600
OS Manufacturer	Microsoft Corporation
System Type	X86-based PC
Processor	x86 Family 15 Model 2 Stepping 4 GenuineIntel 2259 Mhz
Bios Version/Date	IBM 24KT54AUS, 2004. 06.04.
Total Physical Memory	1 024,00 MB
Total Virtual Memory	2,0 GB

Figure 8
Computer description

Conclusions and Future Works

Our novel method is available for compressing the generated weblogs real-time and serves a base for discovering the Internet users and assigning them to real user entities, which is our future plan. We need to deal with the large database, with the fact that cookies may often be deleted therefore a part of them lives too short to be useful for us but needs room in the database. So after the physical compression a logical one will also be needed.

Acknowledgements

This work was accomplished with active cooperation of Median Public Opinion and Market Research Institute and supported by the Mobile Innovation Center, Hungary. Their help is kindly acknowledged.

References

[1] Medán Webaudit, http://www.webaudit.hu/

[2] Coremetrics auditing, http://www.coremetrics.com/technology/first_party.html

[3] CMP United Business Media http://www.cmp.com/delivery/privacy.html

[4] Le Beaumont Language Center http://www.alihk.net/beaumont/clubs/privacy.htm

[5] Csaba Legány, Attila Babos, Sándor Juhász: Cookie-Chain-based Discovery of Relation between Internet Users and Real Persons, 5th International Conference on Information System Development (ISD 2006)

[6] AboutCookies.org, a guide to deleting and controlling cookies www.aboutcookies.org

[7] Cookies: The Perfect User Identification Snack http://www.clickstreamdatawarehousing.com/article06.html