Session Clustering for Faster Internet Access using Rough Set Theory

Haridas Kataria
Board of Technical Education, Haryana

Vipul Pant
Board of Technical Education, Haryana

Abstract – The explosive growth of the web and the increased number of users have led more and more organizations to put their information on the web and provide sophisticated web based services such as social networking, e – leaning, Internet banking, on-line shopping etc. However, the continuous growth in the size and use of the web is increasing the difficulties in managing the web content. Thus, an urgent need exists for developing new techniques in order to improve the web performance. In this context, cluster analysis can be considered as one of the most important aspects in the web mining process for discovering meaningful groups. This article aims to report a technique that involves the identification of the web user sessions and then clustering these sessions based on certain criteria to result into a faster internet access.

Keywords – Web Sessions, RST, Clusters, Web Summary, Prefetching, Caching Value.

I. INTRODUCTION

One of the main issues in web usage mining is the discovery of patterns in the navigational behaviour of Web users (Pallis et al., 2007). Standard approaches do not generally allow characterizing or quantifying the unobservable factors that lead to common navigational patterns (Zamir & Etzioni, 1999). Therefore, it is necessary to develop techniques that can discover hidden and useful relationships among users as well as between users and web – objects. Herein, the focus was laid on identification and clustering of "web user – sessions" as they are defined in Craig Murray et al. (2007). Analyzing the web log and user sessions, the user's behaviour can be understood and the prediction of forthcoming page likely to be accessed by the user can be made. This prediction may then be used to prefetch that page on to the client cache. The methodology used is composed of following steps:

i. Session Identification
ii. RST Implementation
iii. Session Clustering
iv. Clusters Formation

Rough Set Clustering can help make decisions in uncertain situations (Lin & Cercone, 1997). The implementation of user session clustering is done using RST technique and implemented with the help of applications named Session Buddy and Fiddler. The Session Buddy (Mark Wilson & T. Dobrygoski, 2013) and Fiddler (E Lawrence & Fiddler, 2012) are used here to identify the user's sessions and to evaluate the web performance.

II. REVIEW OF LITERATURE

T.Y. Lin and N. Cercone (1997), have reviewed Pawlak's rough set model and its extensions, with emphasis on the formulation, characterization, and interpretation of various rough set models.

Oren Zamir and Oren Etzioni (1999), have observed that the users of web search engines are often forced to sift through the long ordered list of document `snippets' returned by the engines. Further, they have purposed document clustering as an alternative method of organizing retrieval results, but clustering has yet to be deployed on most major search engines.

George Pallis et al. (2006), have claimed that web prefetching is an attractive solution to reduce the network resources consumed by web services as well as the access latencies perceived by Web users. Unlike Web caching, which exploits the temporal locality, Web prefetching utilizes the spatial locality of Web objects. They

presented a clustering-based prefetching scheme where a graph-based clustering algorithm identifies clusters of ‘‘correlated’’ Web pages based on the users’ access patterns.

Premalatha and Natarajan (2007), have considered that the Information Retrieval (IR) is the discipline of searching for documents, for information within documents and metadata about documents. The document clustering improves the retrieval effectiveness of the IR System. If documents can be clustered together in a sensible order and presents a review on document clustering.

Craig Murray et al. (2007), have introduced a novel approach to identify web user sessions based on the burstiness of users’ activity. The method is user-centered rather than population-centered or system-centered and can be deployed in situations in which users choose to withhold personal information.

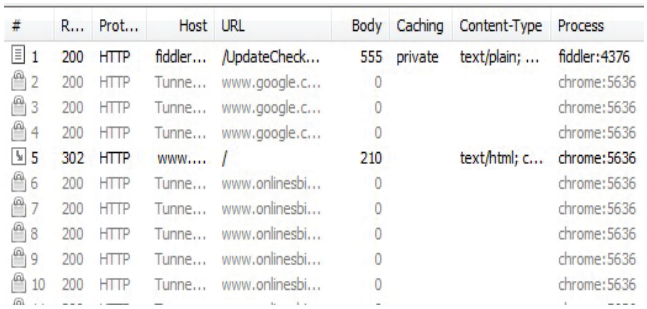
Ling Chen et al. (2007), have been introduced that clustering web users is one of the most important research topics in web usage mining. Existing approaches cluster web users based on the snapshots of web user sessions. The paper did not take into account the dynamic nature of web usage data. In the paper, the focus was on discovering novel knowledge by clustering web users based on the evolutions of their historical web sessions.

Lawrence & Fiddler (2012), purposed a program manager on the internet explorer which captures HTTP and HTTPS traffic and logs it for the user to review It can also be used to modify ("fiddle") with HTTP traffic for troubleshooting purposes as it is being sent or received. By default, traffic from Microsoft's WinINET HTTP(S) stack is automatically directed to the proxy at runtime, but any browser or web application (and most mobile devices) can be configured to route its traffic through Fiddler.

Mark Wilson (2013), developed a new tool, namely as Session Buddy to manage the sessions created. Once installed, it can save and restore sessions and windows from previous browsing sessions. The web browser has a great deal going for it: fast page rendering, a clean interface, powerful extensions etc.

III. USER SESSION IDENTIFICATION

A. Session Identification: A session is the sequence of pages viewed and actions taken by a single user during a defined period of time. A session is defined as a series of related browser requests that come from the same client during a certain time period. The Fiddler, a tool for web debugging, used hare, allows inspecting the traffic, set the breakpoints, and “fiddling" with incoming or outgoing data. The detailed information about the different sessions (like protocol, host, URL used, the caching value, & content type etc.) created during a moment of time is recovered using Fiddler, and is represented by the following fig:



#	R...	Prot...	Host	URL	Body	Caching	Content-Type	Process
1	200	HTTP	fiddler...	/UpdateCheck...	555	private	text/plain; ...	fiddler:4376
2	200	HTTP	Tunne...	www.google.c...	0			chrome:5636
3	200	HTTP	Tunne...	www.google.c...	0			chrome:5636
4	200	HTTP	Tunne...	www.google.c...	0			chrome:5636
5	302	HTTP	www....	/	210		text/html; c...	chrome:5636
6	200	HTTP	Tunne...	www.onlinesbi...	0			chrome:5636
7	200	HTTP	Tunne...	www.onlinesbi...	0			chrome:5636
8	200	HTTP	Tunne...	www.onlinesbi...	0			chrome:5636
9	200	HTTP	Tunne...	www.onlinesbi...	0			chrome:5636
10	200	HTTP	Tunne...	www.onlinesbi...	0			chrome:5636

Fig.1. Web Session summary for a session

Here, the traffic between two web sessions (for example, www.onlinesbi.com & www.espncricinfo.com) is compared for the performance measurement. The performance calculated is given below:

Request Count: 2
Unique Hosts: 2
Bytes Sent: 732 (headers:732; body:0)
Bytes Received: 64,633 (headers:672; body:63,961)
ACTUAL PERFORMANCE

Requests started at: 11:50:02.716

Responses completed at: 11:50:10.763
Sequence (clock) duration: 00:00:08.0464603
Aggregate Session duration: 00:00:05.840
DNS Lookup time: 886ms
TCP/IP Connect duration: 663ms

RESPONSE CODES

HTTP/302: 1
HTTP/200: 1

RESPONSE BYTES (by Content-Type)
---
text/css: 63,751
~headers~: 672
text/html: 210

REQUESTS PER HOST
---
www.onlinesbi.com: 1
www.espncricinfo.com: 1

ESTIMATED WORLDWIDE PERFORMANCE

The following are VERY rough estimates of download times when hitting servers based in WA, USA.
US West Coast (Modem - 6KB/sec)
RTT: 0.20s
Elapsed: 10.20s

Japan / Northern Europe (Modem)
RTT: 0.30s
Elapsed: 10.30s

China (Modem)
RTT: 0.90s
Elapsed: 10.90s

US West Coast (DSL - 30KB/sec)
RTT: 0.20s
Elapsed: 2.20s

Japan / Northern Europe (DSL)
RTT: 0.30s
Elapsed: 2.30s

China (DSL)
RTT: 0.90s
Elapsed: 2.90s

The timeline details of both sessins are given by using the following chart.

B. RST Implementation: A Rough Set, first described by Zdzisùaw I. Pawlak (1982), is a formal approximation of a conventional set in terms of a pair of sets which give the *lower* and the *upper* approximation of the original set. The web log in the form of a Decision Table is treated as rough set for the purpose of clustering of user sessions. Here, the data of a number of sessions accessed during a particular amount of time are

collected is given, which itself explains the characteristics based on which the patterns or clusters are formed.

ID NO	Result Code	Protocol	Host Name	URL	Caching Value
1	200	HTTP	google.com	google.com	3.5
2	200	HTTP	yebhi.com	yebhi.com	3.7
3	302	HTTP	gmail.com	gmail.com	2.6
4	304	HTTP	torentz.com	torentz.com	3.9
5	200	HTTP	hbse.co.in	hbse.co.in	3.3
6	200	HTTP	tunnel to	google.com	2.7
7	302	HTTP	tunnel to	yebhi.com	3.5
8	304	HTTP	tunnel to	gmail.com	3.4
9	200	HTTP	Tunnel to	olx.com	2.9
10	200	HTTP	duddle.com	duddle.com	3.8

Table 1
Decision Table representing the Sessions

In Table 1, the caching values lie in different approximations in which all the assumptions are done purely based on the approximations either lower or upper. Using the theory, the caching values are calculated and represented as follows:

Excellent: caching >= 3.5

Good: 3.5 > caching >=3.0

Average: 3.0 > caching >= 2.5

Three clusters are formed in the range of caching values, i.e. ranging from 2.5 to 3.0 forms a single cluster, from 3.0 to 3.5 forms another cluster and ranging above 3.5 forms another cluster. There are three ranges of cache values, i.e. Excellent, Good and Average caching having their different caching values. Finally, based on these values the, lower and upper approximations are performed. The approximation values are calculated by using the following proposed algorithm.

C. Purposed Algorithm: The following algorithm performs the clustering.

1: Cleaning of web – session's Raw Web Data.

2: If i^{th} value = $(i-1)^{th}$ value, same session exist. Store the Information in the Web Log.

3: Else, check for the current IP address.

4: Check for the Current IP Address is in the Table. If yes, then check for user's IDNo. If user's IDNo is same then check for the Time for the Session. If same or less than a specific period (say 30 min.), the same Session will be there. Else Goto 6.

5: Else, new IP Address will be there. Store the information in the Web Log. Save the session in the session log table.

6: If Time is more than the session – period, then create a new session. The session information will be stored in the Session Log.

7: Now apply the Rough Set Theory for the clustering. As upper approximation & the lower approximation values are available now.

8: Stop the process.

D. Cluster Formation: Using Session Buddy & Fiddler, now by providing the caching values, a number of different clusters are created. After formation of a cluster, by accessing any sessions, the information of all the subsequent sessions will be cached and be available in the same amount of time. Using Session Buddy, the above set of values is used to form the clusters. From the above set of values the following clusters have been formed. The first cluster represents ten no. of sessions (fig 2), whereas the next cluster represents five no. of sessions (fig 3).

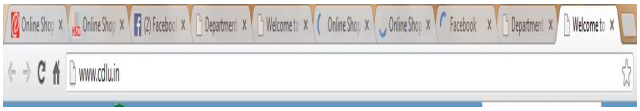


Fig 2 Web User's Session Clusters

Fig 3 Web User's Session Clusters

IV. CONCLUSION

Web user sessions are identified using application Fiddler followed by the application of the rough set theory, wherein the relevant data are grouped in similar and dissimilar patterns. RST algorithm groups the sessions according to their caching value and User_Id. Using the Session Buddy application, these sessions segregated into groups called the Clusters. Finally, these clusters are saved so that they can be easily and quickly accessed during the subsequent request of information cached in clusters, which improves the web performance up to some good extent.

REFERENCES

[1] Zhang, T., Ramakrishnan, R., & Livny, M. (1996). "BIRCH: an efficient data clustering method for very large databases". In ACM SIGMOD, pages 103–114.
[2] Liu, H. & Setiono, R. (1996). "Feature selection and classification – a probabilistic wrapper approach". Proceedings of the 9th International Conference on Industrial and Engineering Applications of AI and ES, 419-424.
[3] Lin, T.Y., Cercone, N., (1997). "Rough sets and Data mining: Analysis of Imprecise Data". Kluwer Academic Publishers.
[4] Shahabi, C., Zarkesh, A., & Shah, V., (1997). "Knowledge discovery from users web-page navigation". In workshop on Research Issues in Data Engineering, England.
[5] Mobasher, B., Cooly, R., & Srivastava, J. (1999). "Automatic personalization based on web usage mining". TR99-010, Department of Computer Science, Depaul University.
[6] Guha, S., Rastogi, R., & Shim, K., (1999). "ROCK: a robust clustering algorithm for categorical attributes". In ICDE.
[7] Shen, Q. & Chouchoulas A., (2002). "A rough-fuzzy approach for generating classification rules". *Pattern Recognition*, 35:2425-2438.
[8] Jensen, R., (2005). "Combining Rough and Fuzzy Sets for Feature Selection", Ph. D thesis, School of Informatics, University of Edinburgh.
[9] Lawrence, Eric, (2005). "Fiddler PowerToy - Part 1: HTTP Debugging ", Microsoft Corporation.
[10] Pallis, George, Vakali Athena, & Pokorny Jaroslav, (2007). "A clustering-based prefetching scheme on a Web cache environment" Department, Faculty of Mathematics and Physics, Charles University, Praha, Czech Republic.
[11] Pillai, G., Girish, (2010). "How to use Fiddler to capture HTTPS sessions and convert to a VSTS Web Test", Microsoft Corporation.
[12] Methews, Lee, (2010). "Session Buddy is a killer session management extension for Google Chrome".
[13] Premalatha, K., & Natarajan, A. M. (2010). "A Literature Review on Document Clustering" Information Technology Journal, 9 : 993 – 1002.
[14] Jones,P., Pearce, C., & Salgueiro, G., (2013). "End-to-End Session Identification in IP-Based Multimedia Communication Networks", Cisco Systems.