ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

Web Usage Mining by Data Preprocessing

[1]Arshi Shamsi, [2]Rahul Nayak, [3]Pankaj Pratap Singh, [4]Mahesh Kumar Tiwari
[1,2,3,4]IIMT Engineering College, Meerut, UP, India

Abstract
Now a day's many web sites are growing and which indirectly leads to increase the complexity of web site designing. To simplify this we have to understand how web sites are being used, how navigation is done, and how many users use it and how much time user spends on pages. This is done using Web Usage Mining. Data sources to usage mining are client side cookies, web server log, software agent etc. This paper presents, how web server log data is preprocesses, which includes data cleaning, user identification and Sessionization, path completion. Once the data is preprocessed it is used for discovering some useful patterns.

Keyword
web usage mining, data cleaning, user identification, Sessionization, path completion.

I. Introduction and Background

[2], Web usage mining is the "automatic discovery of user access patterns from Web servers". Fig. 1 shows the complete process of web usage mining.

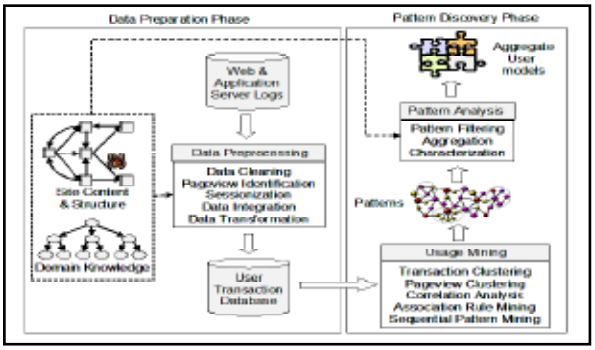


Fig. 1: Web usage mining Process

Data used for Web usage mining, can be collected at one of these three parts [5].

A. Server level collection

The server stores data regarding requests performed by the client, it gives the browsing behavior of the client. All the clickstream are recorded into the server log file. Only information requested by GET is recorded but POST method request is not available on server log.

B. Client level collection

It is the client itself which sends data to a repository. Client-side data collection can be implemented by using a remote agent (such as JavaScript's or Java applets) or by modifying the source code of an existing browser (such as Mosaic or Mozilla) to enhance its data collection capabilities.
Client level collection overcomes the problem of cache data which is not available at server level.

C. Proxy level collection

Proxy level collection sits in between server level and client level. Due to proxy the page load time gets reduce, so user experience high performance.

In this paper, I have cover only the case of a Web server (HTTP server) data.
Following fig. 2 shows the sample web server log data recorded by the WebLog Expert lite tool.

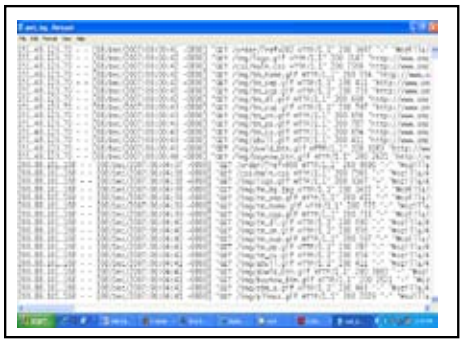


Fig. 2: Web server log data

Web data are classified as follows [5].

1. Usage data

Collected automatically by web server or application servers, which gives the navigational behavior of the user.

2. Content data

This includes the HTML pages, multimedia pages, images, etc.

3. Structure data

Data describing the structure and the organization of the content through internal tags for intra –page or hyper-links for inter-page

4. User profile data

This data is collected when user do registration, purchase some products, do the rating on web site. We can say all client side cookie gives clients data which is used for personalization.
Web usage mining is application of the data mining techniques which uses the various web data sources such as

- Web server log data
- Data about the visitors
- Client side cookies
- Software agent etc.

This data is to be integrated which is very difficult.
Every hit will create a new entry in server log. This server log contains the following fields
151.48.123.71 [08/Dec/2007:00:00:42-0800] "GET /img/logo.gif HTTP/1.1" 200 3147 "http://www.smsync.com/order/?ref=002" "Mozilla/4.0(compatible;MSIE7.0;WindowsNT5.1)" www.smsync.com

- IP address : 151.48.123.71
- Date and time :[08/Dec/2007:00:00:42-0800]
- Method ,file, protocol : GET /img/logo.gif HTTP/1.1
- Status code: 200
- Byte: 3147
- Referrer page : http://www.smsync.com/order/?ref=002
- Agent : Mozilla/4.0(compatible;MSIE7.0;WindowsNT5.1)
- Base URL : www.smsync.com

II. Preprocessing of web log Data

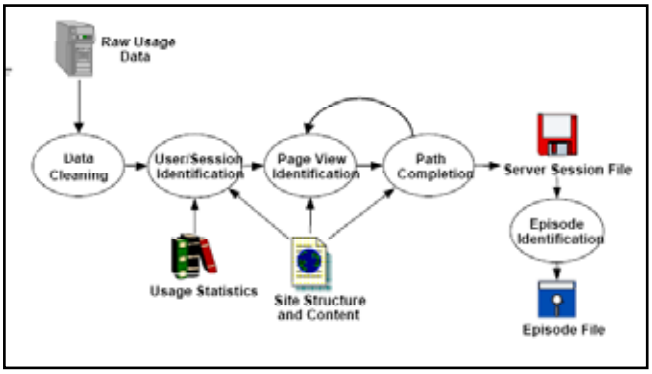


Fig. 3: Preprocessing of web data

[2], Preprocessing include the following steps

1. Data cleaning
2. User identification
3. Session identification
4. Page view matrix /transaction matrix
5. Path completion

In next session we will see this in detail.

III. Data cleaning

This involves removing the unwanted data.

In data cleaning, first the Web server log file is given as input. As for the analysis we required each and every field, so first separate all the fields in web server log file.

Following are the steps for field separation

A. Start

B. Open the logfile

C. Do until the end of file

D. Split the each line belongs to logfile

1. Use space as delimiter to separate the line component.
2. Separated fields are store in variable IP,date, method, referrer, status, byte, agent, url
3. From Date field, brackets are removed and only date, time, gmt is separated using substring method.
4. From method field request , file name and protocol version is separated using substring method
5. From referrer field " "are removed.
6. From agent fields "" are removed and only text part is taken

E. Close the file

Once you have these data, for our analysis we required ip, date, time, method, referrer, agent, url fields. Byte downloaded, status code, and protocol version these fields are not required so we will not consider these fields.

Now in data cleaning phase, unwanted records like image file, style sheets, audio and video files are removed.

If file type is either of .gif, .jpeg ,.css, .au, .wave etc then remove that entry from file variable and all other variables.

86.132.136.211 [08/Dec/2007:00:43:50-0800] "GET /support/key-request.html?ref=104 HTTP/1.1" 200 2580 "-" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; Media Center PC 5.0; .NET CLR 3.0.04506; InfoPath.1)" www.smsync.com

86.132.136.211 [08/Dec/2007:00:43:51-0800] "GET /css/main.css HTTP/1.1" 304 284 "http://www.smsync.com/support/key-request.html?ref=104" "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; Media Center PC 5.0; .NET CLR 3.0.04506; InfoPath.1)"

A. After cleaning the data remains

86.132.136.211 08/Dec/2007 00:43:50 GET /support/key-request.html?ref=104 Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; Media Center PC 5.0; .NET CLR 3.0.04506; InfoPath.1) www.smsync.com

IV. USER Identification

After cleaning data, user identification is done. In user identification, how many users visited the web site is to be identified. This is done by using the ip address and the agent the user has used for browsing the web site [4].

Table 1: Methods for user identification

Method	Description	Privacy Concerns	Advantages	Disadvantages
IP Address + Agent	Assume each unique IP address/Agent pair is a unique user	Low	Always available. No additional technology required.	Not guaranteed to be unique. Defeated by rotating IPs.
Embedded Session Ids	Use dynamically generated pages to associate ID with every hyperlink	Low to medium	Always available. Independent of IP addresses.	Cannot capture repeat visitors. Additional overhead for dynamic pages.
Registration	User explicitly logs in to the site.	Medium	Can track individuals not just browsers	Many users won't register. Not available before registration.
Cookie	Save ID on the client machine.	Medium to high	Can track repeat visits from same browser.	Can be turned off by users.
Software Agents	Program loaded into browser and sends back usage data.	High	Accurate usage data for a single site.	Likely to be rejected by users.

We use the first method. Fig. 4 shows the process of user identification.

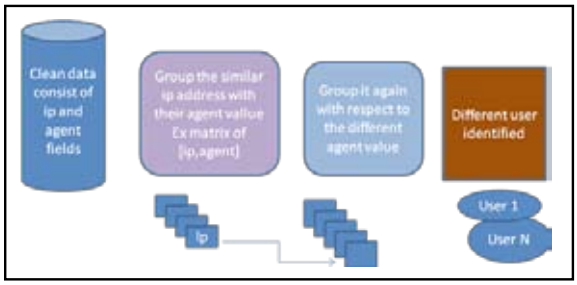


Fig. 4: User identification processes

Web server log data alone will not give correct user identification because many users are using the one IP address because of proxy server. Otherwise one user can have several IP address when the load balancing is done by ISP on different proxies.

But still it is not accurate; we can use the client side user's cookies. But possibility is that user may disable the cookies

Some problems we faced in user identification are [5]

A. Single IP address/Multiple Server Sessions

Internet service providers (ISPs) typically have a pool of proxy servers that users access the Web through. A single proxy server may have several users accessing a Web site, over the same time period.

B. Multiple IP address/Single Server Session

Some ISPs randomly assign each request from a user to one of several IP addresses. In this case, a single server session can have multiple IP addresses.

C. Multiple IP address/Single User

A user that accesses the Web from different machines will have a different IP address from session to session. This makes tracking repeat visits from the same user difficult.

D. Multiple Agent/Singe User

Again, a user that uses more than one browser, even on the same machine, will appear as multiple users. Assuming each user has now been identified (through cookies, logins, or IP/agent/path analysis), the clickstream for each user must be divided into sessions.

Since page requests from other servers are not typically available, it is difficult to know when a user has left a Web site.

Fig. 5 example shows how user identification done using first method.

Time	IP	URL	Ref	Agent
0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:10	2.3.4.5	C	-	IE6;WinXP;SP1
0:12	2.3.4.5	B	C	IE6;WinXP;SP1
0:15	2.3.4.5	E	C	IE6;WinXP;SP1
0:19	1.2.3.4	C	A	IE5;Win2k
0:22	2.3.4.5	D	B	IE6;WinXP;SP1
0:22	1.2.3.4	A	-	IE6;WinXP;SP2
0:25	1.2.3.4	E	C	IE5;Win2k
0:25	1.2.3.4	C	A	IE6;WinXP;SP2
0:33	1.2.3.4	B	C	IE6;WinXP;SP2
0:58	1.2.3.4	D	B	IE6;WinXP;SP2
1:10	1.2.3.4	E	D	IE6;WinXP;SP2
1:15	1.2.3.4	A	-	IE5;Win2k
1:16	1.2.3.4	C	A	IE5;Win2k
1:17	1.2.3.4	F	C	IE6;WinXP;SP2
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

User 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

User 2

0:10	2.3.4.5	C	-
0:12	2.3.4.5	B	C
0:15	2.3.4.5	E	C
0:22	2.3.4.5	D	B

User 3

0:22	1.2.3.4	A	-
0:25	1.2.3.4	C	A
0:33	1.2.3.4	B	C
0:58	1.2.3.4	D	B
1:10	1.2.3.4	E	D
1:17	1.2.3.4	F	C

Fig. 5: Example of user identification

V. Sessionization

Once we have, identified the user, now focus on each and every user's activity i.e. finds the sessions. Session is the time between the logged in and logged out. During these time user visits/ click streams many pages. Session is to find the sequence of clickstream and trace the user activity. User may visit the same web site many times. So there is more than one session for one user [9].

Session reconstruction = correct mapping of activities to different individuals + correct separation of activities belonging to different visits of the same individual [4]

Sessions are constructed using time based heuristics and navigation based heuristics methods.

A. Time based heuristics

1. H1: Total session duration time should not exceed a maximum (i.e30 minutes). Given t0, the timestamp for the first request in a constructed session S, the request with a timestamp t is assigned to S, if and only if t-t0<=30 min.
2. H2 ; Total page time must not exceed the maximum (10 minute). Given t1, the timestamp for request assigned to constructed session S, the next re-quest with timestamp t2 is assigned to S, if and only if t2-t1<=10 min

B. Navigation based heuristics

H-ref: a page must have been reached from the referrer page in the same session. If the referrer page is not invoked from same sessions consider new session.

But the problem is there for the undefined referrer pages. If the page is start page or the page is visited via back button then referrer page is not defined, it is recorded as "-". So it is consider as new session which is actually not [9-10].

User 1

Time	IP	URL	Ref
0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Session 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C

Session 2

1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Fig. 6: Using h1 and h2 heuristics

VI. Path completions

Next important task is path completions. Sessionization will give you the sequence of the user activity but still some of the references are missing due to the caching.

E.g the user return the page A which is in its current sessions if that page is cached at client side then no request is made to server and hence no enttry at the server log present.

This missing references is found by the knowledge of web site Structrure and the referrer information available on server log. Fig. 7, shows the missing references.

The structure of a site is created by the hypertext links be between page views. The structure can be obtained and preprocessed in the same manner as the content of a site. A different site structure may have to be constructed for each server session.

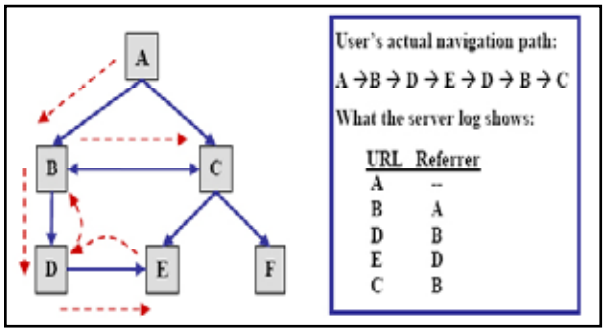


Fig. 7: Missing references using path completion

VII. Conclusion

In this paper I have presented the complete web data preprocessing required for web usage mining by considering the server log. Web data is clean, User identification is done by considering the IP address and agent used. After that sessions are identified. In Sessionization missing references are present, which are removed using path completion method. Next work to be considered is transaction identification .These preprocessed data is then used for finding the useful patterns. Future work is to do a pattern analysis on this preprocessed data and the techniques as per described in [5], are:

A. Statistical analysis

This technique is used to do traffic analysis for web site as well to find most visited pages, average daily hits, etc.;

1. Association rules

Find the frequently occurred page view in the transaction which satisfies the minimum support. Also used to group the most often reference pages together in one session.

2. Sequential patterns

The attempt of this technique is to discover time ordered Sequences of URLs followed by past users, in order to predict future ones. This may used to place the advertisement for the certain users.

3. Clustering

Some meaningful clusters of URLs can be created by discovering similar characteristics between them according to user's behaviors. This kind of information is most useful for segmenting the market for e-business and for the web search engines.

References

[1] B. Mobasher, N. Jain, E. Han, J. Srivastava,"Web mining: Pattern discovery from world wide web transactions", Technical Report TR 96-050, University of Minnesota, Dept. of Computer Science, Minneapolis, 1996.

[2] R. Cooley, B. Mobasher, J. Srivastava,"Grouping web page references into transactions for mining world wide web browsing patterns", Technical Report TR 97-021, University of Minnesota, Dept. of Computer Science, Minneapolis, 1997.

[3] Miha Grčar,"Web usage mining:user profiling".

[4] "Web Usage Mining for E-Business Applications", ECML/ PKDD-2002 Tutorial, 19 August 2002.

[5] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang-Ning Tan,"WebUsage Mining: Discovery and applications of usage patterns from Web data", 2000.

[6] Robert Cooley, Bamshad Mobasher, Jaideep Srivastava ,"Web Mining: information and pattern discovery on the World Wide Web", 1997.

[7] Robert Cooley, Bamshad Mobasher, Jaideep Srivastava, "Data preparation for mining World Wide Web browsing patterns", 1999.

[8] Gabriele Bartolini,"Web usage mining and discovery of association rules from HTTP servers logs", 8th oct. 2001.

[9] "A Framework for the Evaluation of Session Reconstruction Heuristics in Web Usage Analysis".

[10] "Web usage mining by Bamshad Mobasher pdf".



Arshi Shamsi received his B.Tech. degree in computer science from Vira College of Engineering,(from U.P.T.U. Lucknow) Bijnor, India, in 2004 and pursuing M.Tech. degree in Computer Science from Alfalah School of Engineering and Technology,(Maharishi Dayanand University, Rohtak) Haryana. He is working as Assistant Professor, with Department of Computer Science, at IIMT Engineering College. He is having a total work experience of 7 years.



Rahul Nayak received his B.Tech. degree in computer science & Engg from Raj Kumar Goel Institute of Technology, Ghaziabad, India, in 2006 and pursuing M.Tech. Degree in Computer Science from IITT Engg. College, Pojewal (Punjab Technical University, Jalandhar). He is working as Assistant Professor, with Dept. of Computer Science, at IIMT Engg. College. He is having a total work experience of 5 years.



Pankaj Pratap Singh received his B. Tech degree in Computer science from Uttar Pradesh Technical University, Lucknow, India, in 2007 and M.Tech degree in Medical Image and Image Processing from Indian Institute of Technology Kharagpur, Kharagpur, India, in 2010. He is working an Assistant Professor, IIMT Meerut, Meerut, India. His research interests include digital signal processing, image processing techniques, computer network and data mining technique.



Mahesh Kumar Tiwari received his B.Tech. degree in computer science from B.I.T.,(from U.P.T.U. Lucknow) Meerut, India, in 2007 and pursuing M.Tech. degree in Computer Science from IIMT Engineering College,(from M.T.U. Noida) Meerut. He is working as Assistant Professor, with Department of Information Technology, at IIMT Engineering College. He is having a total work experience of 4 years.