Neural Network Approach for Web Usage Mining

Ketki Muzumdar, Ravi Mante, Prashant Chatur

Abstract— Web usage mining attempts to discover useful knowledge from the secondary data obtained from the interactions of the users with the Web. Web usage mining has become very critical for effective Web site management, business and support services, personalization, and network traffic flow analysis and so on. Previous study on Web usage mining using a concurrent Clustering, Neural based approach has shown that the usage trend analysis very much depends on the performance of the clustering of the number of requests. In this paper, a novel approach Self Organizing Map is introduced, which is a kind of neural network, in the process of Web Usage Mining to detect user's patterns. We are going to analyze the traditional K-Means algorithm result with comparison to SOM. The process details the transformations necessaries to modify the data storage in the Web Servers Log files to an input of SOM.

Index Terms— Clustering, K-Means, SOM, Web Server Log File, Web Usage Mining.

I. INTRODUCTION

Web mining is the application of data mining techniques to extract knowledge from Web data including Web documents, hyperlinks between documents, usage logs of web sites, etc.. A panel organized at ICTAI 1997 asked the question "Is there anything distinct about Web mining (compared to data mining in general)?" While no definitive conclusions were reached then, the tremendous attention on Web mining in the past five years, and a number of significant ideas that have been developed, have answered this question in the affirmative in a big way. In addition, a fairly stable community of researchers interested in the area has been formed, largely through the successful series of WebKDD workshops, which have been held annually in conjunction with the ACM SIGKDD Conference since 1999, and the Web Analytics workshops, which have been held in conjunction with the SIAM data mining conference [1].

Web Mining can be broadly divided into three distinct categories, according to the kinds of data to be mined:

- *Web content mining:*

Web Content Mining is the process of extracting useful information from the contents of Web documents. Content data corresponds to the collection of facts a Web page was designed to convey to the users. It may consist of text, images, audio, video, or structured records such as lists and tables.

Text mining and its application to Web content has been the most widely researched. Some of the research issues addressed in text mining are, topic discovery, extracting association patterns, clustering of web documents and classification of Web Pages. Research activities in this field also involve using techniques from other disciplines such as Information Retrieval (IR) and Natural Language Processing (NLP). While there exists a significant body of work in extracting knowledge from images – in the fields of image processing and computer vision – the application of these techniques to Web content mining has not been very rapid.

- *Web structure mining:*

The structure of a typical Web graph consists of Web pages as nodes, and hyperlinks as edges connecting between two related pages. Web Structure Mining can be regarded as the process of discovering structure information from the Web.

Web usage mining: Web Usage Mining is the application of data mining techniques to discover interesting usage patterns from Web data, in order to understand and better serve the needs of Web-based applications. Usage data captures the identity or origin of Web users along with their browsing behavior at a Web site. The usage data can also be split into three different kinds on the basis of the source of its collection: on the server side, the client side, and the proxy side. The key issue is that on the server side there is an aggregate picture of the usage of a service by all users, while on the client side there is complete picture of usage of all services by a particular client, with the proxy side being somewhere in the middle.

II. WEB USAGE MINING

The aim of Web usage mining is to discover patterns of user activities in order to better serve the needs of the users for example by dynamic link handling, by page recommendation etc. The aim of a Web site or Web portal is to supply the user the information which is useful for him. There is a great competition between the different commercial portals and Web sites because every user means eventually money (through advertisements, etc.). Thus the goal of each owner of a portal is to make his site more attractive for the user. For this reason the response time of each single site have to be kept below 2s. Moreover some extras have to be provided such as supplying dynamic content or links or recommending pages for the user that are possible of interest of the given user. Clustering of the user activities stored in different types of log files is a key issue in the Web community.

Revised Manuscript Received on 30 May 2013.

* Correspondence Author
Ketki Muzumdar, Department of Computer Science & Engineering, Govt. College of Engineering Amravati , Amravati, India.
Ravi Mante, Department of Computer Science & Engineering, Govt. College of Engineering Amravati , Amravati, India.
Dr. Prashant Chatur , Department of Computer Science & Engineering, Govt. College of Engineering Amravati , Amravati, India.

There are three types of log files that can be used for Web usage mining. Log files are stored on the server side, on the client side and on the proxy servers. By having more than one place for storing the information of navigation patterns of the users makes the mining process more difficult. Really reliable results could be obtained only if one has data from all three types of log file. The reason for this is that the server side does not contain records of those Web page accesses that are cached on the proxy servers or on the client side. Besides the log file on the server, that on the proxy server provides additional information. However, the page requests stored in the client side are missing. Yet, it is problematic to collect all the information from the client side. Thus, most of the algorithms work based only the server Side data[2]. Web usage mining consists of four main steps:

A. Data collection
B. Preprocessing,
C. Pattern discovery
D. Pattern analysis

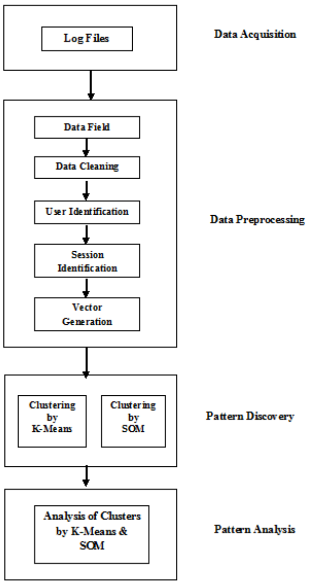

Fig 1. Model for Neural Network Approach for Web Usage Mining.

In the preprocessing phase the data have to be collected from the different places it is stored (client side, server side, proxy servers). After identifying the users, the click-streams of each user has to be split into sessions. In general the timeout for determining a session is set to 30 minute. The pattern discovery phase means applying data mining techniques on the preprocessed log data.

It can be frequent pattern mining, association rule mining or clustering. In this paper we are dealing only with the task of clustering web usage log. In web usage mining there are two types of clusters to be discovered: usage clusters and page clusters. The aim of clustering users is to establish groups of users having similar browsing behavior. The users can be clustered based on several information. In the one hand, the user can be requested filling out a form regarding their interests, for example when registration on the web portal. The clustering of the users can be accomplished based on the forms. On the other hand, the clustering can be made based on the information gained from the log data collected during the user was navigating through the portal. Different types of user data can be collected using these methods, for example (I) characteristics of the user (age, gender, etc.), (ii) preferences and interests of the user, (iii) user's behavior pattern. The aim of clustering web pages is to have groups of pages that have similar content. This information can be useful for search engines or for applications that create dynamic index pages. The last step of the whole web usage mining process is to analyze the patterns found during the pattern discovery step. Web Usage Mining try to understand the patterns detected in before step. The most common techniques is data visualization applying filters

High dimensional data stream contains a tremendous huge amount of data. Such massive amount data contains a large data with high dimensions with data complexity. For example wireless sensor network data, web logs, Google search, etc. Traditional methods are not suitable over high dimensional data as they required very high computation cost for processing data.

III. PROBLEM DEFINITION

A. Web log structure

The log files are text files that can range in size from 1KB to 100MB, depending on the traffic at a given a website. The dataset of “Government College of engineering and technology, Amravati.” is used for the process. This data contains a record of user interactions with the college website (originally http://www.gcoea.ac.in).

117.203.75.190 - - [10/Sep/2012:18:01:43 +0530] "GET /prajwalan2012/css/events.css HTTP/1.1" 200 1085 "http://www.gcoea.ac.in/prajwalan2012/" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.2 (KHTML, like Gecko) Chrome/15.0.861.0 Safari/535.2"

There are various fields in this dataset are:

- *IP address:* "117.203.75.190 "

This is the IP address of the machine that contacted our site.

- *Username etc:* "- -"

Only relevant when accessing password-protected content.

- *Timestamp:* [10/Sep/2012:18:01:43 +0530]”

Time stamp of the visit as seen by the web server.

- *Access request:*"GET/prajwalan2012/css/events.css HTTP/1.1"

The request made. In this case it was a "GET" request (i.e. "show me the page") for the file “/cgi-bin/log/source/vs/vs_main.cgi” using the "HTTP/1.1" protocol. A "HEAD" request fetches only the document header, and is the web equivalent of a "ping" to check your page is still there and hasn't changed.

- *Result status code:* “200”

The resulting status code. “200” is success. If the requested URL didn’t exist, this is where the dreaded “404” would have shown up in the log.

- *Bytes transferred:* "1085"

The number of bytes transferred. If this matches the size of the file requested, so this is a successful download. If the number is less, then that would indicate a failed or partial download. Some user agents can fetch files a bit at a time. Each bit will show up as a separate line in the log file, so a series of "hits" whose total adds up to, or exceeds, the file size could indicate a successful download. On the other hand it could indicate someone having trouble connecting to site who has to keep reconnecting.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

- *ReferrerURL:*“http://www.gcoea.ac.in/prajwalan2012/"

The referring page. Not all user agents supply this information. This is the page the visitor is on when they clicked to come to this page. Sometimes this is simply the page the user was looking at when they typed in address into their browser, or clicked on the address in some other software such as a newsreader or an email client. This information is very useful to webmasters, as it allows them to measure which sites are driving traffic to their site. It also represents a small loss of privacy, as it lets us see where visitors are coming from.

- *User agent:*"Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.2 (KHTML, like Gecko) Chrome/15.0.861.0 Safari/535.2"

The "User Agent" identifier. The User Agent is whatever software the visitor used to access this site. It's usually a browser, but it could equally be a web robot, a link checker, an FTP client or an offline browser. The "user agent" string is set by the software manufacturer, and can be anything they choose to be. In this case "Mozilla/4.7" probably means Netscape 4.7, "[en]" probably implies it's an English version, "Win 95" indicates Windows 95 etc, etc. Well-behaved web bots and spiders will usually use this string to identify themselves, their web site and an email address.

B. Data preparation

There are some important technical issues that must be taken into consideration during this phase in the context of the Web personalization process, because it is necessary for Web log data to be prepared and preprocessed in order to use them in the consequent phases of the process[5].

The data pre-processing consist of the following steps:

- *Data field extraction:*

The log file is read character by character up to the end and then by using the methods of String Tokenizer class the data fields are broken into tokens and saved in an array. Before data storage we need to create the table named as log table in which each entry from the original log file is stored. Total 4210 records are stored in database.

- *Data Cleaning:*

With the important entries, a web log file may consist of certain undesirable rather useless data which has nothing to do with the mining procedure. Data cleaning concerned with removing all the data tracked in web logs that are useless for mining purposes e.g. requests for graphical page content (e.g. jpg, jgeg, gif, js, css, swf, avi, mov, etc.). Request for any other file which might be included in the web page or even navigation sessions performed by robots and web spiders. Robots and web spider navigation patterns must be explicitly identified. After the cleaning process the records are reduced to around 50% i.e. 2710 records.

- *User identification:*

Users is uniquely Identified by combination of referrer URL and user agent (eg. 1.2.3.4 + IE5; Win2k).There are 175 unique users in 2710 records.

- *User session:*

After identify the users; we need to identify the sessions. To do this we can divide the access of the same users in sessions. It’s difficult to detect when one session is finish and start another. To detect sessions is common use of time between requests; if two requests are called in of time frame, we can suppose that these requests are in the same session; in other way below of time frame we can consider two different sessions. A good time frame is 30 minutes.

- Vector generation:

After session Identification, they will be represented in the form of vectors. The Urls will be represented by their corresponding weights in the particular session. The weight will be frequency of occurrence of that Url in a session. The vector representation is given below will be given as input to our clustering algorithms.

C. Pattern discovery

The pattern discovery phase means applying data mining techniques on the preprocessed log data. It can be frequent pattern mining, association rule mining or clustering.

D. Pattern analysis

The pattern analysis phase means analyzing the pattern with respect to different performance criteria such as time, Squared error, Correlation etc.

IV. SOLUTION PROPOSED

Given the preprocessing steps outlined above, for the rest of this implementation we assume that there is a set of n unique URLs U = {url1, url2, …,urln}, appearing in the preprocessed log, and a set of m user transactions T= {t1, t2, …, tm}, where each ti ϵ T is a non-empty subset of U.

$$t = <w (V_1^{(A)}, t), w (V_2^{(A)}, t)\ldots w (V_n^{(A)}, t)>$$

In this paper clustering is done and for that two algorithm k-Means and Self Organizing Map (SOM) is used. The artificial neural network, in this case SOM, has an arbitrary number of input neurons, this number is pre-defined , to do this the number of the most common pages in the site; by the way each site is probably has different artificial neural network architecture. The output of SOM is a map of M X N dimensions; the user configures N; this is the number of N cluster that the users want to obtain; in the output only one cluster will be activated. The same pattern of input will generate the activation of the same output cluster; similar inputs will be activated near output clusters.

V. RESULT

The algorithm above will be compared in different criteria on gcoea.ac.in web site logs. The resulting winner algorithm will be used for designing an application of recommendation system.

The criteria are as follows:

1. Varying number of cases
2. Time required by the algorithms while varying number of cases.
3. Different number of clusters i.e. considering different values of k and number of output nodes in K-Means and Self Organizing Map algorithm and Sum of Squared Error (SSE) respectively.

Analysis of the clusters formed by both algorithms considering above conditions with respect to percentage of occurrence of each unique url in corresponding clusters. This paper; we can see the comparison between both methods, SOM and K-Means, in gcoea.ac.in sites.

For this site it has been develop a complete process involved in Web Usages Mining. This website contains information about college, student , department , staff etc. .The log file consist of one month data total 71,238 lines and 2045 unique IPs. After extracting and cleaning the information from the complicated log file, logs are divided into 30 min sessions. Total 2621 sessions, which are further represented by vector to provide input to the algorithms. The Urls will be represented by weight of url in that session. The weights will be the frequency of occurrence of url in the sessions.

The weight vector are still not ready to pass as input to the algorithms, as they contain sessions which do not have observable effect on analysis of the algorithm on the contrary these sessions increases execution complexity due to long feature vector length . The method of filtering is called collaborative filtering. After the process, cases reduces to 684 and feature vector length reduces to 43. The result is analyze through the tables given below.

At first we applied different number of cases and compare the time required for both algorithms to execute.

# Cases	K-Means (Secs)	SOM (Secs)
228	4.63	0.38
456	7.32	0.40
684	16.15	0.67

Table.1. Comparison with respect to time of execution of K-Means and SOM

We can observe that as the number of cases increases Self Organizing map has better performance than K-Means with respect to time. With different number of cases and different number of clusters, next criteria will be considered *i.e.* Sum of Squared Error (SSE). The SSE is calculated by first evaluating Square error within each cluster which is the sum of square of distance between points in that cluster and its centre. Given as:

$$e_k^2 = \sum_{i=1}^{n_k} (x_{ik} - M_k)^2 \qquad (1)$$

M_k is the mean vector of cluster C_K defined as its centroid and given by

$$M_k = ({}^{1}/_{n_k}) \sum_{i=1}^{n_k} x_{ik} \qquad (2)$$

Where x_{ik} is the i^{th} sample belonging to cluster C_K. The square-error for cluster C_K is the sum of the squared Ecludian distance between each sample in C_K and its centroid. This error is also called the within-cluster variation. The square-error for the entire clustering space containing K cluster is the sum of the within-cluster variations

$$E_k^2 = \sum_{k=1}^{K} e_k^2 \qquad (3)$$

By evaluation the above criteria by changing number of clusters, we can observe from the flowing table that SOM out stand with respect to SSE for all the considered values of clusters. But in table 2 we can observe that with small number of cases and clusters K-Means works great, but as the number increases performance of K-Mean degrades.

Cluster #	K-Means (SSE)	SOM (SSE)
3	7017	321
4	5263	243
5	4210	195
6	3508	162

Table.2. Comparison with respect to SSE with different # clusters and # cases of K-Means and SOM

VI. CONCLUSION

Analyzing the outcome of the algorithms, we can conclude that with respect K-Means we can cover more Urls but SOM works better for larger number of cases. With increase in data, learning process of SOM becomes more accurate and we can consider larger number of clusters. SOM is also efficient in time as compared to K-Means. Thus we can conclude that SOM has better performance than K-Mean. For more accurate result we can provide output of K-Means as a input to SOM and use the results of SOM for generating recommendation System and analysis of the Web site.

Cluster #	# Cases by K-Means	# Cases by SOM
	# Cases 228	
3	13:215:0	228:0:0
4	11:214:3:0	228:0:0:0
5	24:2:0:183:9	228:0:0:0:0
6	10:213:4:0:0:1	228:0:0:0:0:0
	# Cases 456	
3	16:35:405	434:7:15
4	2:17:23:414	6:10:434:6
5	11:434:1:0:10	6:26:415:3:6
6	1:32:2:130:291:0	5:24:9:2:414:2
	# Cases 684	
3	21:41:622	635:8:41
4	20:38:3:623	41:635:2:7
5	1:41:10:121:511	625:16:1:3:37
6	0:237:11:1:425:0	625:16:2:1:3:37

Table.3. Comparison with respect to different # clusters and # cases of execution of K-Means and SOM

REFERENCES

1. R. Kosala, H. Blockeel, “Web Mining Research: A Survey”, SIGKKD Explorations, vol. 2(1), July 2000.
2. Magdalini Eirinaki , Michalis Vazirgiannis, “Web Mining for Web Personalization”, ACM Transactions on Internet Technology, Vol. 3, No. 1, February 2003.

3. J. Srivastava, R. Cooley, M. Deshpande, P.-N. Tan, “Web Usage Mining: Discovery And Applications Of Usage Patterns From Web Data”, SIGKKD Explorations, vol.1, Jan 2000.
4. Vinita Shrivastava, “Web Usage Data Clustering Using Neural Network Learning”, IJRIM Vol. 1, No. 2 , June, 2011.
5. Navin Kumar Tyagi, A.K. Solanki& Sanjay Tyagi, “An Algorithmic Approach To Data Preprocessing In Web Usage Mining” International Journal of Information Technology and Knowledge Management, Vol. 2, No. 2, July-December 2010, pp.: 279-283.
6. Masseglia, F., Poncelet, P., And Cicchetti, R. (1999). “WebTool: An integrated framework for data mining”, In Proceedings of the Ninth International Conference on Database and Expert Systems Applications (DEXA’99) (Florence, Italy, August1999, pp.: 892–901.
7. Spiliopoulou, M. And Faulstich, L. C.. “WUM: A web utilization miner”, Proceedings of the International Workshop on the Web and Databases (Valencia, March) 1998.
8. Perkowitz, M. And Etzioni, O. 2000. “Towards adaptive web sites: Conceptual framework and case study”, In Artif. Intell. 118, 1–2,pp.: 245–275.
9. Mobasher, B., Dai, H., Luo, T., Sung, Y., And Zhu, J. 2000c. “Integrating web usage and content mining for more effective personalization”, In Proceedings of the International Conference on Ecommerce and Web Technologies (ECWeb2000). (Greenwich, UK, Sept.).
10. Paola Britos, Damián Martinelli, Hernán Merlino, Ramón García-Martínez, “Web Usage Mining Using Self Organized Maps”, IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.6, June 2007.
11. A.M. Mora, C.M. Fernandes, J.J. Merelo, V. Ramos, J.L.J. Laredo, A.C. Rosa, “Kohonants: A Self-Organizing Ant Algorithm For Clustering And Pattern Classification”, Artificial Life XI 2008.
12. Santhi, S.Shrivasan.P ,“An improved Usage Mining using Back Propagation Algorithm With Functional Update” , Advance computing Conference, IACC 2009.
13. Prakash S Raghavendra, Shreya Roy Chowdhury, Srilekha Vedula Kameswari, “Web Usage Mining Using Statistical Classifiers And Fuzzy Artificial Neural Networks”, International Journal Multimedia and Image Processing (IJMIP), Volume 1, Issue 1, March 2011.
14. A. Jirayusakul, S. Auwatanamongkol, “A Supervised Growing Neural Gas Algorithm for Cluster Analysis”, International Journal of Hybrid Intelligent Systems 3 2006.

AUTHOR PROFILE

Ketki Muzumdar received her B.E. degree in Information Technology from Sipna’s college of Engineering and Technology, Amravati, Maharashtra, India, in 2010, pursuing M.tech. degree in Computer Science and Engineering from Government college of Engineering, Amravati, Maharashtra, India. Her research interests include web data mining, artificial neural network. At present, She is engaged in Neural Network approach for web usage mining.

Prof. Ravi V. Mante received his B.E. degree in Computer science and Engineering, from Government college of Engineering, Amravati, Maharashtra, India in 2006, the M.tech. degree in Computer science and Engineering, from Government college of Engineering, Amravati, Maharashtra, India, in 2011. He is Assistant professor in Government college of Engineering, Amravati, Maharashtra, India. from 2007. His research interests include ECG signal analysis, soft computing technique, cloud computing. At present, He is working with Artificial neural network.

Dr. P N Chatur has received his M.E. degree in Electronics Engineering from Govt. College of Engineering Amravati, India and Ph.D degree from Amravati University. He Has Published twenty papers in national Conferences and Ten papers in international journals. His area of research includes Artificial Neural Network, Data Mining, Data Stream Mining and Cloud computing. Currently he is Head of Computer Science and Engineering Department at Govt. College of Engineering Amravati, Maharashtra India. At present he is engaged with large database mining analysis and stream mining.