International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Web Usage Mining for Comparing User Access Behavior using Clustering

Amit Dipchandji Kasliwal[1], **Dr. Girish S. Katkar**[2]

[1]Inter Institutional Computer centre, Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur- 440023, Maharashtra, India

[2] Dept. of Computer Science, Arts, Sci., Commerce College, Koradi, Nagpur, Maharashtra, India

Abstract: ***Motivated by the practical needs in improving the access to information over the internet with comparison between the users, many authors have given their most time on studying and designing the framework for the application of data mining like web mining, text mining and so on. Now it is the time to study data mining in systematic manner with few foundational measures related to the implementation of web mining through proposed low level prototype WebUMining. In this study, authors worked on data mining techniques applied successfully for suggesting the web usage mining to the organizational and authority to improve their website by comparing the user access and access behavior with result discussed and analysis. Authors used datasets from reliable repositories, which applied with algorithm proposed in WebUMining model. These datasets treated as training data for the all proposed algorithms. This study of data mining has primarily focused on the mining algorithms which are used in web usage mining.***

Keywords: Web Usage Mining, User Access Behavior, Clustering, Web Access Log Data, WEKA.

1. Introduction

Web mining has been implemented and developed into several research areas to satisfy the needs of website developer and trend analysts. By Etzioni et al [1], the term web mining is prepared to symbolize the use of data mining techniques for automatically discovering hidden information from the data available over internet in the form of web content, website structure and web usage. The research paper explains the algorithm based on predictive approach of data mining technique used for web usage mining for comparing access behavior. When applied to web access log data for comparing user access behavior, clustering technique used to group the accesses based on the characteristics. The main advantage that Clustering has it makes group of common characteristics like user"s interest, behavior together to classify different user requests. In this research work, authors proposed an algorithm named ClusterView based on clustering technique.

Clustering can be supervised data mining [2] or unsupervised data mining as given in [3] means that it does not rely on predefined training itemset or semisupervised data mining proposed in [4]. Clustering can be of two types either distance based or of model based. In distance based clustering technique, it involves determining distance computed between pairs of data items and then creating group of similar items putting together into clusters such as partitioned clustering and hierarchical clustering. Partitioned methods create partitions of data items into *K* groups and it is represented by k-means algorithm. A hierarchical method used to build hierarchical group (or set) of interleaved clusters with the top level cluster containing a single cluster of all data item. In model based clustering defined by Zhong et al and Ghosh et al [5], the model type is proposed as the model structure that can be determined by model selection techniques and parameters estimated using maximum likelihood algorithms such as Expectation Maximization (EM) technique. In proposed ClusterView algorithm, authors used partitioned and hierarchical clustering method to compare the user access behavior on internet. Clustering can also be used in data preprocessing task for classification technique that could operate on the detected clusters.

2. Implementation of Proposed Algorithm

In the proposed algorithm, implementation is approaches towards a representation of the process analyzed by clustering in order to discover patterns that used for preparing comparison model based on the characteristic of user who accessed the particular website with respect to attributes of the user showing the interest or attributes of the context in which the activities taken place.

The first is more relevant in this scenario, the focus is on the identification of need by experts while developing the website. In proposing the algorithm, authors prepared clustering model implemented with WebUMining prototype in the combination with simulated monitored data that usually relates to the activities themselves. The proposed technique involves a dynamic manner clustering process controlled in order to function towards different goals. Thus, we not only have to determine these goals but additionally also need to establish metrics that can be used for this kind of dynamic control. This is the purpose of the clustering model to provide data sets containing a certain number of data items. The analysis process varies as required for visualization, complying with the respective clustering goal. In implementation of proposed ClusterView algorithm the authors studied the following key aspects that put together in the implementation.

a. Entropy is often associated with the certainty or uncertainty as discussed in Tribus et al and McIrvine et al in[7] it is assigning to an event with the order or disorder in a system which is, however, discussed controversially. Although a universal definition for sure of the term entropy not exist as there is common valuation in its implications and interpretation. Here, entropy based clustering technique used to determine the distribution of particular items into clusters. An item in this context can be either a user or an event.

b. Variance: The main goal behind clustering techniques is to maximize the distances between multiple clusters and to minimize the distances within clusters. For achieving this goal, authors made the provision by setting the variance of attribute at low value in a good cluster setting. This implies that similar values for multiple attributes are found in the same clusters.

3. Proposed ClusterView Algorithm

To implement the ClusterView algorithm, authors applied partitioning algorithm to finds groups of strongly related (or correlated) pages by making partition according to its connected items those have the common characteristics that used for comparing the web access behavior using web access log data stored at server side. At first proposed algorithm performs Depth First Searching techniques in the web access log file that brings all related component in to matrix M by searching for the file. Once the item found, the algorithm checks if there are any items which not get visited or considered. It means that a related item has been split and therefore it needs to be identified to get into M. To repeat it, the searching technique again applied by starting from one of item that not visited. Proceeding further, in the proposed system authors used the two key aspects as discussed earlier, these are entropy and variance which are implemented in the proposed algorithm by counting the time spend on requestedURL and the occurrence in the manner of the number of users used a particular page. These two considerations used in identifying the relation between each pair of webpages. Therefore authors proposed a formula to bring all those pages together which are accessed maximum number of times with the maximum number of time spent on it. This formula is developed with statistical experiments that used the harmonic mean of time spent and the occurrence foe each page as given below:

$$Support_i = \frac{2 * (TimeTaken_i\ *\ Occurrence_i)}{(TimeTaken_i\ +\ Occurrence_i)}$$

Where,

$Support_i$ is calculated minimum support measure for webpage i
$TimeTaken_i$ is time taken by user that spent on accessing webpage i
$Occurrence_i$ is the total number of time that page i is accessed

Let us take this formula for counting support of each webpage in the web access log file. Considering a worst case, when all the URLs are in the same cluster, the output generated by the algorithm will have execution cost of this algorithm it the linear form of data available in the web access log data. Before putting items into the clusters for getting navigational pattern profile, the generated clusters are then ranked based on data available in the matrix M.

3.1 Pseudo code and Flowchart

The proposed algorithm at first scans each web access log entry from web access log one after and then stores it with assigning occurrence of each item to 0. In this context, the item is the webpage that accessed by user during particular session. These items at first get stored in the list *L*, while reading the items from itemset, the matrix *M* with items that satisfies the formula as discussed above. This matrix *M* is then loaded with items those are present in the web log file with the time taken by user to spend over webpage and the occurrence of that web page. After completing the data satisfying formula from web access log data, this item in matrix *M* is get assigned to the current item for cluster *C*. After looking at each entry for item into matrix *M*, then next is to scan for the items those not satisfies the defined formula for minimum support. For making clusters with respect to the user defined minimum support count taken as input. Perhaps, the items with support that not matches the item available in the matrix *M* are then get searched with the help of depth first searching technique so that it returns the item which are below the user defined minimum support and then algorithm creates cluster for this items. If the occurrence for an item goes more than defined support, this element belongs to the cluster where cluster size is given by user as an input for which website administrator is looking for.

The proposed algorithm is provided with the following pseudo code hence the authors are intends to have the implementation of the proposed algorithm in the low level prototype model.

Algorithm Name: **ClusterView**
Input: Web access log file after data cleaning and preprocessing.
Minimum Support measure
Minimum Cluster Size
Output: List of Clusters.

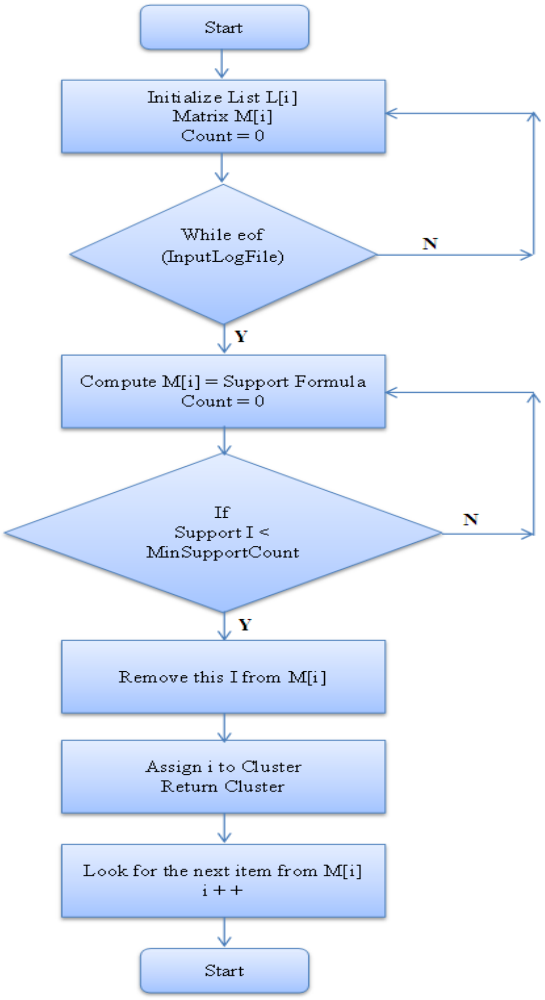
Table 1: Proposed ClusterView Algorithm

```
1.  Begin
2.  Initialize List L with storing each log record into L[i]
    Initialize Matrix M
    Initialize count = 0
    Whileeof (InputFile)
    For webpage i ∈ List L[i]
        Do computation of matrix M [i] = Formula of Support for i
        count = M [i]
    End For
    End While
3.  For each i in matrix M[i]
    If Support i<MinimumSupportCount
        Remove this i from M[i]
    End If
    End For
4.  For each i ∈ M[i]
        Do Cluster C[i1]= DFS (i);
    If C[i] <MinimumClusterSize
        Remove this i from C[i]
    End If
    i = i + 1
    End For
5.  Return C with i
6.  End
```

While implementing the proposed ClusterView algorithm, initially there was no method to consider a target page that can be used as centered for the cluster while comparing the user access and to make the cluster depending upon the common characteristics. And hence, the most accessed page taken as the target page. The output of the proposed ClusterView algorithm is in the form of a text file containing the list of clusters with respect to the support measure given as input from the user. In this format the results are then be easily represented by visualization software. Effectiveness of the proposed ClusterView algorithm is implemented and evaluated through comparing processing time and number of patterns generated during the web access log data submitted as input to algorithm so that it can be useful for making comparative analysis of the data.

3.2 Flowchart

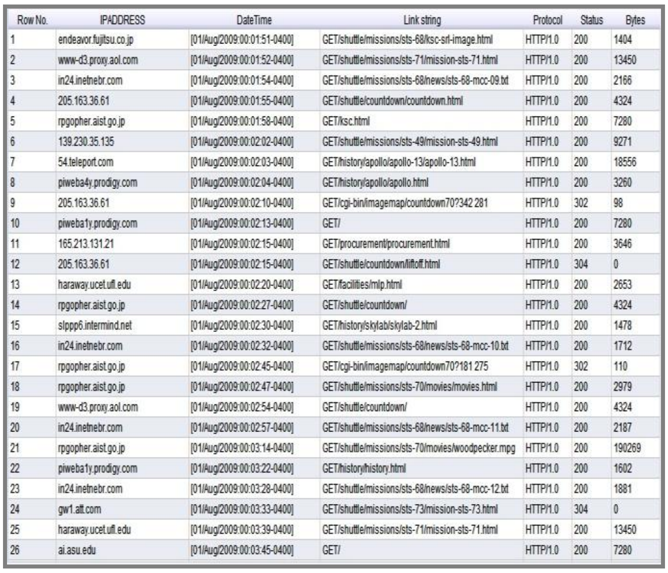
Following is the flowchart of the proposed ClusterView algorithm as discussed above. The flowchart contains all the keywords that discussed during implementation of the algorithm discussed earlier. The flowchart gives the idea about the flow of data and the actual process implemented in the proposed ClusterView algorithm. The algorithm is then implemented with JAVA and it is one of the algorithms of WebUMining model used for web usage mining for comparing user access behavior.



Flowchart of proposed ClusterView Algorithm

4. Visualizing Result

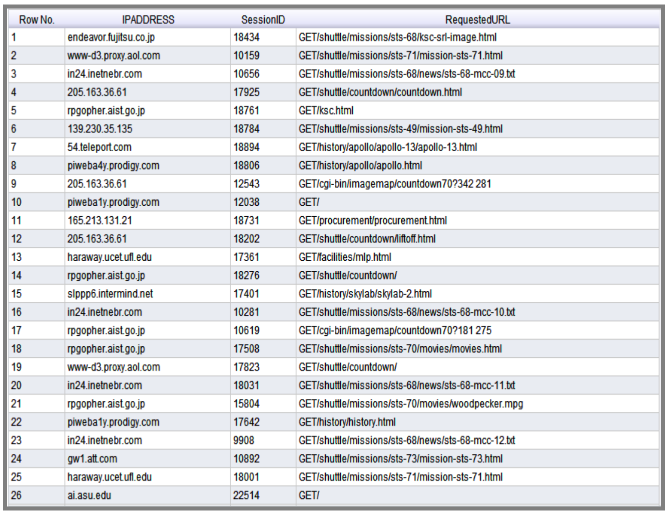
In this section, authors worked on the web access log data „NASALog.arff‟ of NASA‟s Kennedy Space Centre‟s website available at the archive.org. On this input file, experiment performed to find out the clusters base on the web pages and the maximum times spend over that webpage. The proposed ClusterView algorithm can also be used to compare and for predicting user access behavior through web access log data and it is implemented in WebUMining model. During experiment, the focus is on looking and searching for the webpages that are accessed by the users who had the common characteristics about their behavior access from web log data. Then this file is converted from text file to arff file format using WEKA as shown in figure 1 given below:



Row No.	IPADDRESS	DateTime	Link string	Protocol	Status	Bytes
1	endeavor.fujitsu.co.jp	[01/Aug/2009:00:01:51-0400]	GET/shuttle/missions/sts-68/ksc-srl-image.html	HTTP/1.0	200	1404
2	www-d3.proxy.aol.com	[01/Aug/2009:00:01:52-0400]	GET/shuttle/missions/sts-71/mission-sts-71.html	HTTP/1.0	200	13450
3	in24.inetnebr.com	[01/Aug/2009:00:01:54-0400]	GET/shuttle/missions/sts-68/news/sts-68-mcc-09.txt	HTTP/1.0	200	2166
4	205.163.36.61	[01/Aug/2009:00:01:55-0400]	GET/shuttle/countdown/countdown.html	HTTP/1.0	200	4324
5	rpgopher.aist.go.jp	[01/Aug/2009:00:01:58-0400]	GET/ksc.html	HTTP/1.0	200	7280
6	139.230.35.135	[01/Aug/2009:00:02:02-0400]	GET/shuttle/missions/sts-49/mission-sts-49.html	HTTP/1.0	200	9271
7	54.teleport.com	[01/Aug/2009:00:02:03-0400]	GET/history/apollo/apollo-13/apollo-13.html	HTTP/1.0	200	18556
8	piweba4y.prodigy.com	[01/Aug/2009:00:02:04-0400]	GET/history/apollo/apollo.html	HTTP/1.0	200	3260
9	205.163.36.61	[01/Aug/2009:00:02:10-0400]	GET/cgi-bin/imagemap/countdown70?342 281	HTTP/1.0	302	98
10	piweba1y.prodigy.com	[01/Aug/2009:00:02:13-0400]	GET/	HTTP/1.0	200	7280
11	165.213.131.21	[01/Aug/2009:00:02:15-0400]	GET/procurement/procurement.html	HTTP/1.0	200	3646
12	205.163.36.61	[01/Aug/2009:00:02:15-0400]	GET/shuttle/countdown/liftoff.html	HTTP/1.0	304	0
13	haraway.ucet.ufl.edu	[01/Aug/2009:00:02:20-0400]	GET/facilities/mlp.html	HTTP/1.0	200	2653
14	rpgopher.aist.go.jp	[01/Aug/2009:00:02:27-0400]	GET/shuttle/countdown/	HTTP/1.0	200	4324
15	slppp6.intermind.net	[01/Aug/2009:00:02:30-0400]	GET/history/skylab/skylab-2.html	HTTP/1.0	200	1478
16	in24.inetnebr.com	[01/Aug/2009:00:02:32-0400]	GET/shuttle/missions/sts-68/news/sts-68-mcc-10.txt	HTTP/1.0	200	1712
17	rpgopher.aist.go.jp	[01/Aug/2009:00:02:45-0400]	GET/cgi-bin/imagemap/countdown70?181 275	HTTP/1.0	302	110
18	rpgopher.aist.go.jp	[01/Aug/2009:00:02:47-0400]	GET/shuttle/missions/sts-70/movies/movies.html	HTTP/1.0	200	2979
19	www-d3.proxy.aol.com	[01/Aug/2009:00:02:54-0400]	GET/shuttle/countdown/	HTTP/1.0	200	4324
20	in24.inetnebr.com	[01/Aug/2009:00:02:57-0400]	GET/shuttle/missions/sts-68/news/sts-68-mcc-11.txt	HTTP/1.0	200	2187
21	rpgopher.aist.go.jp	[01/Aug/2009:00:03:14-0400]	GET/shuttle/missions/sts-70/movies/woodpecker.mpg	HTTP/1.0	200	190269
22	piweba1y.prodigy.com	[01/Aug/2009:00:03:22-0400]	GET/history/history.html	HTTP/1.0	200	1602
23	in24.inetnebr.com	[01/Aug/2009:00:03:28-0400]	GET/shuttle/missions/sts-68/news/sts-68-mcc-12.txt	HTTP/1.0	200	1881
24	gw1.att.com	[01/Aug/2009:00:03:33-0400]	GET/shuttle/missions/sts-73/mission-sts-73.html	HTTP/1.0	304	0
25	haraway.ucet.ufl.edu	[01/Aug/2009:00:03:39-0400]	GET/shuttle/missions/sts-71/mission-sts-71.html	HTTP/1.0	200	13450
26	ai.asu.edu	[01/Aug/2009:00:03:45-0400]	GET/	HTTP/1.0	200	7280

Input web access log data file for ClusterView algorithm

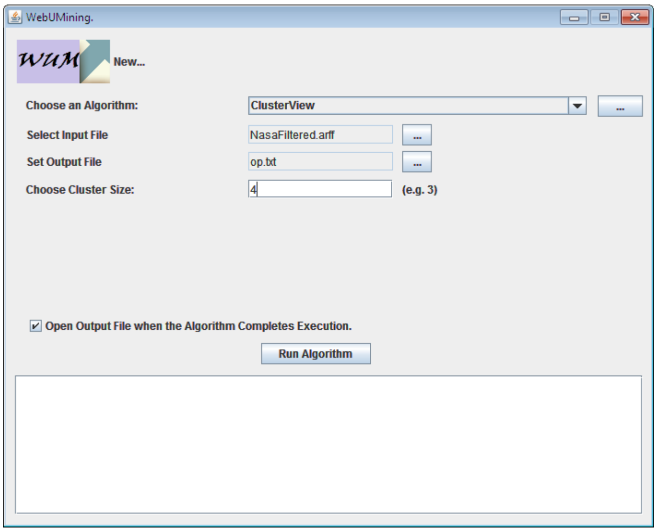
The log file produced as shown in the figure 6.1is then filtered so that it only contains the fields which are IPaddress, sessionID considered from the webpage time which is accessed maximum number of time by the users and the webpage that requested maximum number of times as requested URL. Figure 6.2 shows the filtered file prepared with WEKA containing the fields as discussed above.



Row No.	IPADDRESS	SessionID	RequestedURL
1	endeavor.fujitsu.co.jp	18434	GET/shuttle/missions/sts-68/ksc-srl-image.html
2	www-d3.proxy.aol.com	10159	GET/shuttle/missions/sts-71/mission-sts-71.html
3	in24.inetnebr.com	10656	GET/shuttle/missions/sts-68/news/sts-68-mcc-09.txt
4	205.163.36.61	17925	GET/shuttle/countdown/countdown.html
5	rpgopher.aist.go.jp	18761	GET/ksc.html
6	139.230.35.135	18784	GET/shuttle/missions/sts-49/mission-sts-49.html
7	54.teleport.com	18894	GET/history/apollo/apollo-13/apollo-13.html
8	piweba4y.prodigy.com	18806	GET/history/apollo/apollo.html
9	205.163.36.61	12543	GET/cgi-bin/imagemap/countdown70?342 281
10	piweba1y.prodigy.com	12038	GET/
11	165.213.131.21	18731	GET/procurement/procurement.html
12	205.163.36.61	18202	GET/shuttle/countdown/liftoff.html
13	haraway.ucet.ufl.edu	17361	GET/facilities/mlp.html
14	rpgopher.aist.go.jp	18276	GET/shuttle/countdown/
15	slppp6.intermind.net	17401	GET/history/skylab/skylab-2.html
16	in24.inetnebr.com	10281	GET/shuttle/missions/sts-68/news/sts-68-mcc-10.txt
17	rpgopher.aist.go.jp	10619	GET/cgi-bin/imagemap/countdown70?181 275
18	rpgopher.aist.go.jp	17508	GET/shuttle/missions/sts-70/movies/movies.html
19	www-d3.proxy.aol.com	17823	GET/shuttle/countdown/
20	in24.inetnebr.com	18031	GET/shuttle/missions/sts-68/news/sts-68-mcc-11.txt
21	rpgopher.aist.go.jp	15804	GET/shuttle/missions/sts-70/movies/woodpecker.mpg
22	piweba1y.prodigy.com	17642	GET/history/history.html
23	in24.inetnebr.com	9908	GET/shuttle/missions/sts-68/news/sts-68-mcc-12.txt
24	gw1.att.com	10892	GET/shuttle/missions/sts-73/mission-sts-73.html
25	haraway.ucet.ufl.edu	18001	GET/shuttle/missions/sts-71/mission-sts-71.html
26	ai.asu.edu	22514	GET/

Input web access log data file after filtering

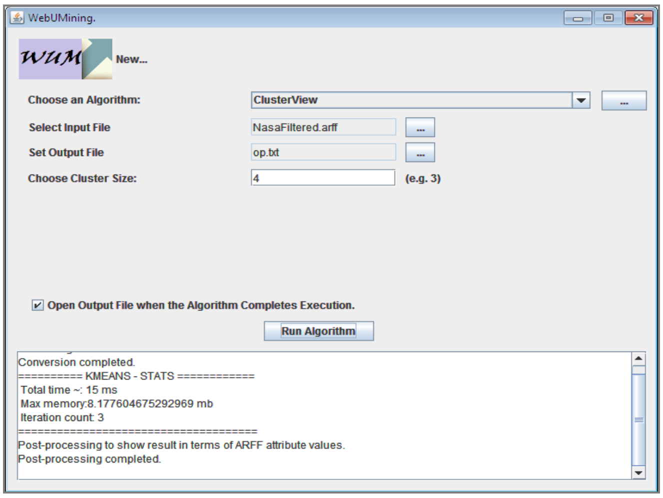
The developed model named WebUMining accepts this input file and after selecting the proposed ClusterView algorithm that performs its process as discussed earlier. Overall process of the algorithm implemented with the WebUMining model is represented with the following figures.



Choose an Algorithm: ClusterView
Select Input File: NasaFiltered.arff
Set Output File: op.txt
Choose Cluster Size: 4 (e.g. 3)

☑ Open Output File when the Algorithm Completes Execution.

WebUMining UI for selecting ClusterView Algorithm



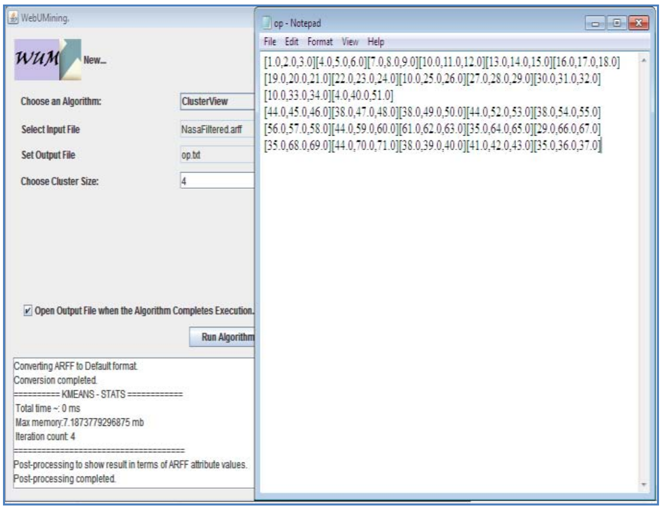
Choose an Algorithm: ClusterView
Select Input File: NasaFiltered.arff
Set Output File: op.txt
Choose Cluster Size: 4 (e.g. 3)

☑ Open Output File when the Algorithm Completes Execution.

```
Conversion completed.
========== KMEANS - STATS ============
Total time ~: 15 ms
Max memory:8.177604675292969 mb
Iteration count: 3
=====================================
Post-processing to show result in terms of ARFF attribute values.
Post-processing completed.
```

WebUMining UI for selecting ClusterView Algorithm

Result from the ClusterView algorithm achieved as shown and the visualization is represented using the same technique as authors visualized the AssociationRule algorithm to represent the associated webpages. With the minimum cluster size 4 in that the numbers represent the webpages that are accessed by the users and we can simply compare these after looking towards the each cluster combined in between [ ] (square brackets) as shown. The web pages that form part of a specific cluster are then Listed with the specific cluster as shown in class with some colored similar to show their class.



Choose an Algorithm: ClusterView
Select Input File: NasaFiltered.arff
Set Output File: op.txt
Choose Cluster Size: 4

☑ Open Output File when the Algorithm Completes Execution.

```
Converting ARFF to Default format.
Conversion completed.
========== KMEANS - STATS ============
Total time ~: 0 ms
Max memory:7.1873779296875 mb
Iteration count: 4
=====================================
Post-processing to show result in terms of ARFF attribute values.
Post-processing completed.
```

```
[1.0,2.0,3.0][4.0,5.0,6.0][7.0,8.0,9.0][10.0,11.0,12.0][13.0,14.0,15.0][16.0,17.0,18.0]
[19.0,20.0,21.0][22.0,23.0,24.0][10.0,25.0,26.0][27.0,28.0,29.0][30.0,31.0,32.0]
[10.0,33.0,34.0][4.0,40.0,51.0]
[44.0,45.0,46.0][38.0,47.0,48.0][38.0,49.0,50.0][44.0,52.0,53.0][38.0,54.0,55.0]
[56.0,57.0,58.0][44.0,59.0,60.0][61.0,62.0,63.0][35.0,64.0,65.0][29.0,66.0,67.0]
[35.0,68.0,69.0][44.0,70.0,71.0][38.0,39.0,40.0][41.0,42.0,43.0][35.0,36.0,37.0]
```

Result of WebUMining using ClusterView algorithm

After getting the results with respect to the user‟s common characteristics based on the user access the research is showing the clusters that makes each access to the requested URL from multiple users during particular session.

Table 2: Cluster Based on the IPAddress

<table>
<tr><th colspan="2">Clusters Based on IPAddress</th></tr>
<tr><td colspan="2">Time taken to build model (full training data) : 122 seconds
Clustered Instances
<b>0 9 (32%)</b>
<b>1 19 (68%)</b>

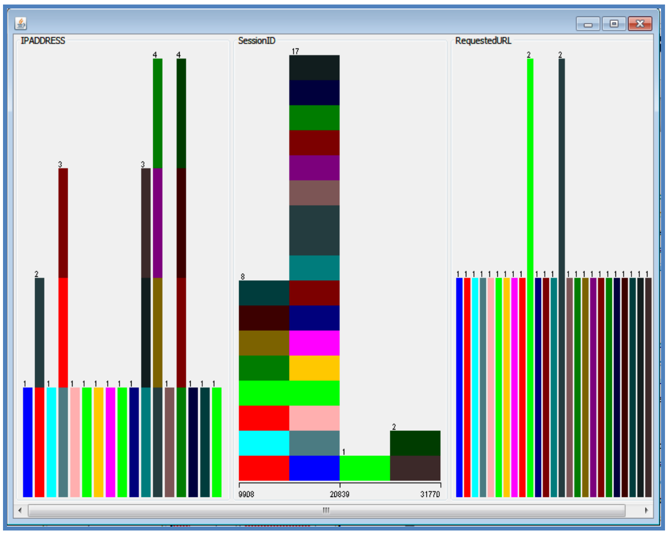
Class attribute: IPADDRESS
Classes to Clusters:
0 1 <-- assigned to cluster</td></tr>
<tr><td>0 1 | endeavor.fujitsu.co.jp
1 1 | wwwd3.proxy.aol.com
1 0 | in24.inetnebr.com
1 2 | 205.163.36.61
0 1 | rpgopher.aist.go.jp
0 1 | 139.230.35.135
0 1 | 54.teleport.com
0 1 | piweba4y.prodigy.com
1 0 | piweba1y.prodigy.com</td><td>0 1 | 165.213.131.21
0 3 | haraway.ucet.ufl.edu
1 3 | rpgopher.aist.go.jp
0 1 | slppp6.intermind.net
2 2 | in24.inetnebr.com
0 1 | piweba1y.prodigy.com
1 0 | gw1.att.com
1 0 | ai.asu.edu</td></tr>
<tr><td colspan="2"><b>Cluster 0</b><-- in24.inetnebr.com
<b>Cluster 1</b><-- haraway.ucet.ufl.edu</td></tr>
</table>

Table 2 shows the cluster for IPAddress that spend the maximum time on the particular page. The cluster 0 represents the common items containing the IPAddresses of the user who followed a sequential path and during that they made the visits to more than one webpage in a session. While Cluster two shows the class of those IPAddresses who visited the webpages at least once during surfing the internet. During comparison between the clusters depending upon the access made by them is represented in the manner of percentage against the total percentage of users who accessed the webpages. This clusters represented with the numbers 0 and 1 so that it will be easy for website administrator to compare the users those are accessing the website more than once and creating sequential navigational path during access to website. The items in cluster 0 contain 32% of the overall access made to the website while cluster 1 contains 68% total access.

Table 3: Cluster Based on the Requested URL

Clusters Based on Requested URL	
Time taken to build model (full training data) : 217 seconds Clustered Instances **0 11 (39%)** **1 17 (61%)** Class attribute: RequestedURL Classes to Clusters: 0 1 <-- assigned to cluster	
0 1 | GET/shuttle/missions/sts-68/ksc-srl-image.html 1 0 | GET/shuttle/missions/sts-71/mission-sts-71.html 1 0 | GET/shuttle/missions/sts-68/news/sts-68-mcc-09.txt 0 1 | GET/shuttle/countdown/countdown.html 0 1 | GET/ksc.html 0 1 | GET/shuttle/missions/sts-49/mission-sts-49.html 0 1 | GET/history/apollo/apollo13/apollo-13.html 0 1 | 54.teleport.com 0 1 | piweba4y.prodigy.com 1 0 | piweba1y.prodigy.com 0 1 | GET/shuttle/technology/sts-newsref/sts_asm.html 1 0 | GET/shuttle/missions/sts-68/news/sts-68-mcc-13.txt 0 1 | GET/shuttle/missions/sts-71/mission-sts-71.html 1 0 | GET/shuttle/missions/sts-73/mission-sts-73.html	0 1 | GET/history/apollo/apollo.htm 1 0 | GET/cgi-in/imagemap/countdown70?342 281 1 1 | GET/ 0 1| GET/procure/procurement.htn 0 1 | GET/shuttle/countdown/liftoff.htn 0 1 | GET/facilities/mlp.html 0 2 | GET/shuttle/countdown/ 0 1 | GET/history/skylab/skylab-2.html 1 0 | GET/shuttle/missions/sts-68/news/sts-68-mcc-10.txt 1 0 | GET/cgi-bin/imagemap/countdown70?181 275 0 1 | GET/shuttle/missions/sts-70/movies/movies.html 1 0 | GET/shuttle/missions/sts-68/news/sts-68-mcc-11.txt 1 0 | GET/shuttle/missions/sts-70/movies/woodpecker.mpg 0 1 | GET/history/history.html 1 0 | GET/shuttle/missions/sts-68/news/sts-68-mcc-12.txt
Cluster 0<-- GET/shuttle/missions/sts-71/mission-sts-71.html **Cluster 1**<-- GET/shuttle/countdown/	

Table 3 represents the clusters created on RequestedURL information of the web access log file. The cluster 0 represents the class of those webpages which are accessed more than once while cluster 1 represents the class that contains the webpages those are accessed at least once. Figure 6.6 shows the groups of the filtered information which used for clustering as discussed and shown in the figure 6.1. The visualization of these information and the generated results are shown in the following figures.



Result of WebUMining using ClusterView algorithm

The figures given below are the graphical representation of the clusters created with the proposed ClusterView and visualizes with the WEKA software for interpretation. Figure A shows the cluster with respect to IPAddress information whereas Figure B shows the cluster with respect to the RequestedURL information from the filtered web access log data.

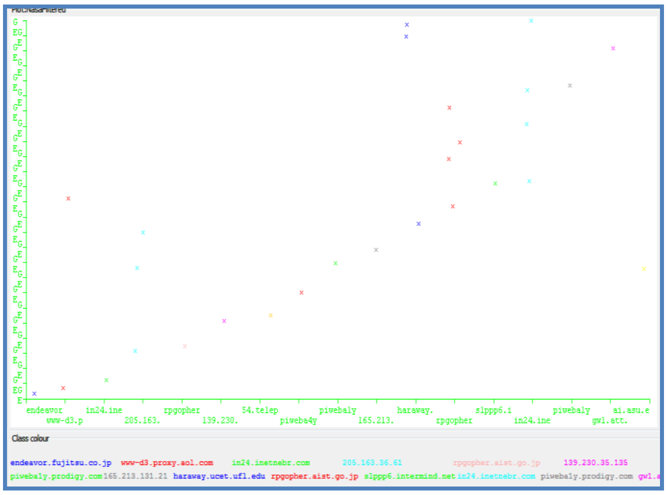


Figure A: Visualization of Cluster based on IPAddress

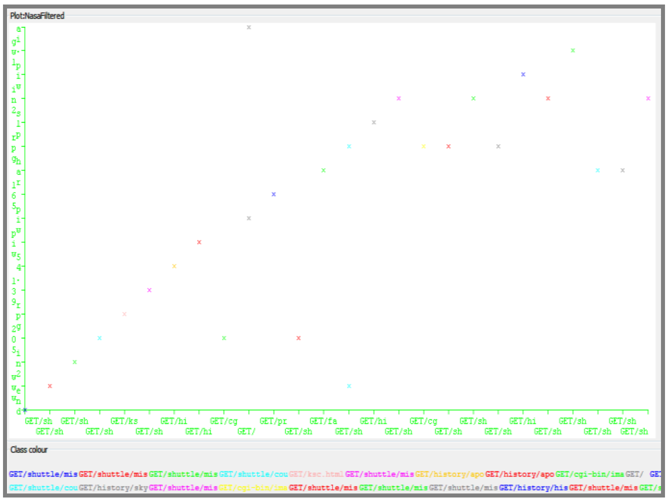

Plot:NasaFiltered
GET/sh GET/sh GET/ks GET/hi GET/cg GET/pr GET/fa GET/hi GET/cg GET/sh GET/hi GET/sh GET/sh
GET/sh GET/sh GET/sh GET/hi GET/ GET/sh GET/sh GET/sh GET/sh GET/sh GET/sh GET/sh GET/sh
Class colour
GET/shuttle/mis GET/shuttle/mis GET/shuttle/mis GET/shuttle/cou GET/ksc.html GET/shuttle/mis GET/history/apo GET/history/apo GET/cgi-bin/ima GET/ GET
GET/shuttle/cou GET/history/sky GET/shuttle/mis GET/cgi-bin/ima GET/shuttle/mis GET/shuttle/mis GET/shuttle/mis GET/history/his GET/shuttle/mis GET/s

Figure B: Visualization of Cluster based on RequestedURL

The result can be interpreted and determined by the requirement criteria when given as inputs after selecting ClusterView algorithm. At the initial stage of the visualization, it involves the visualization software and then for specific form result authors used WEKA and SPSS software for same. These software provides the advantage of being accessible for any process of analysis over log data hence it makes sensible for making decision about the access to internet so that for any type of comparison since it is in demand to use for web usage mining process.

5. Recommendations and Limitation

The aim of this research work is in using the proposed WebUMining model in order to suggest the comparison regarding the user access to the particular web site. In order to do this, WebUMining model used to process the web usage data from three different repositories to determine the changes or improvements needed to made a particular webpage accessible to the user. This web access log data is processed with ClusterView algorithm. In proposed clustering based ClusterView algorithm, authors recommending that if the website is design, build and implemented with what the user want and what link the user accessing then website administrator can made that improvements in their website so that comparing user will make the access easy to the website.

A limitation of WebUMining is that no support is provided for automatically recommendation of a particular web site depending upon the user access and on other hand by comparing the user access behavior, web administrator cannot take decision whether to provide the next webpage in the path or not through website design. As the web site structure is dynamic in the form of there hierarchy of the webpage attached to one another and hence due to the newly attached link varies the accesses to the different webpages those involves in sequential path and this creates the result that cannot give the comparison. For this variation in the input information, the comparison based on user‟s behavior over website needs to have sufficient domain knowledge regarding website while access based user behavior being compared and analyzed.

Conclusion

This research work proposed by the authors demonstrate a low level prototype called WebUMining which implemented with the proposed ClusterView Algorithm which can be used to compare the user access behavior using web access log data through data mining techniques. The proposed algorithm is implemented using Java and WEKA. For visualization of the result we have used the SPSS and RapidMiner software.

References

[1] Perkowitz M. and Etzioni O., “Adaptive websites: Conceptual cluster mining”, International Joint Conference on Artificial Intelligence, 1999.
[2] Eick C., Zeidat N. and Zhao Z., “Supervised clustering algorithms and benefits”, International Conference on Tools with Artificial Intelligence, 2004.
[3] Thomas F. and Joachims T., “Supervised k-Means Clustering”, 2003.
[4] Albanese M., Picariello A. and Sansone C., “A Web Personalization System based on Web Usage Mining Techniques”, ACM WWW, 2004.
[5] Basu S., Bilenko M. and Mooney R., “A probabilistic framework for semi-supervised clustering”, ACM SIGKDD, 2004.
[6] Zhong S. and Ghosh J., “A Unified Framework for Model-based Clustering”, Journal of Machine Learning Research, 2003.
[7] Tribus M. and McIrvine E., "Energy and Information", Scientific America, 1971.
[8] Valsamidis S., Kontogiannis S., Kazanidis I., Theodosiou T. and Karakos A, “A Clustering Methodology of Web Log Data for Learning Management Systems”, Educational Technology & Society, 2012.
[9] Guo J., Keselj V. and Gao Q., “Integrating Web Content Clustering into Web Log Association Rule Mining”, Advances in Artificial Intelligence, 2005.
[10] Etzioni O., “The World Wide Web: Quagmire or gold mine”, Communications of the ACM, 1996.
[11] Rajimol, “Data Mining For Web Intelligence”, Thesis, MG University, 2013.
[12] http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html