# Clustering Techniques for the Identification of Web User Session

**Nirmala Huidrom[*], Neha Bagoria[**]**

[*] Department of Computer Science and Engineering, Jodhpur Institute of Engineering and Technology
[**] Department of Computer Science and Engineering, Jodhpur Institute of Engineering and Technology

***Abstract***- The web user-session can be defined as a set of several TCP connections generated by a single user while surfing the web during a given time frame. An activity period, i.e. session, is terminated by a long silent period. This activity period is comprised of several TCP connections which may be used to transfer data. However, identification of active and silent period is not trivial. Correct identification of session is the main goal of our study. Traditional method used threshold-based mechanism for the identification of web user-sessions which required a priori definition of the threshold value. This method is very sensitive to the threshold value, which is very difficult to set correctly. By using clustering techniques, web user-sessions can be identified without requiring a priori definition of threshold values. This paper is based on the definition and identification of web user-sessions. The main goal of this paper is to exploit the property of clustering techniques to group TCP connections in order to identify web user sessions and to compare the performance with that of the threshold-based mechanism.

***Index Terms*** **–** Clustering method, session identifications, similarity measurement, web server log.

## I. INTRODUCTION

Nowadays, World Wide Web (www) becomes a powerful platform to distribute and to retrieve information. They can be accessed by companies, governments, universities, students, teachers, businessmen and some users. Because of the rapid growth of the web, the field related to web becomes an important research area. One such research area includes retrieval of useful information from the web server log. Web server log is a text file created by web server which record activity of users on the servers. Log file come in several different formats. One such format looks like this:

128.159.122.137 - - [01/Aug/1995:11:01:04 -0400] "GET /finance/main.htm HTTP/1.0" 200 1974.

Each record contains IP address, username, timestamp, access request, result status code and bytes transferred. Log file may contain one to ten thousands records of requests every day. To retrieve the useful information from this large amount of information, log entries should be grouped into session by using some techniques. The most commonly used technique for identifying session is the timeout method. In this method, if the inter-arrival time between two TCP connections is smaller than the pre-defined threshold value then the connections are part of the same session and if the inter-arrival time is larger than the threshold value than the second connections is the first element of a new session. The main limitation of this method is that they are very sensitive to the threshold value and required a priori definition of the threshold value. If the threshold value is not correctly set then the following errors may occurs:

- If the threshold value is too large then the condition for combining two unrelated connections in the same session may arrive.
- If the threshold value is too small then the condition for separating two related connections into two different sessions may arrive.

To overcome the limitations of the threshold-based mechanism, most researchers used different clustering techniques to identify web user-sessions. Informally, clustering is the process of dividing the given data into groups of similar objects in such a way that objects in the same cluster are similar to each other but different from objects of other clusters. The clustering is also known as cluster analysis, data clustering, segmentation analysis, unsupervised classification or taxonomy analysis. Clustering is widely used in numerous applications such as pattern recognition, data analysis, image processing etc.

The main aim of this paper is to exploit the property of clustering techniques to identify the web user sessions. Performance is compared with that of the threshold based mechanism and concludes that algorithm that used clustering technique is more robust than the threshold based mechanism.

The remainder of this paper is organized as follows: First, the section, related work, contains a brief overview of the methods that have been applied till date for the identification of web user session. Then, in the next section, clustering techniques is described in detail. The method for identifying web user session by using clustering techniques is explained in the section "Using clustering

techniques on web server log". In the section, performance analysis, the performance of the algorithm is presented and compared it with that of the threshold based mechanism. The last section of this paper provides the conclusions and future works.

## I. RELATED WORK

There are several definitions and session identification methods reported in the literature. In [1], web user session is defined based on the four stages: submission of query, IR component invocation, selection and navigation and reformation. In [2], web user session is defined as a set of TCP connections created by a given user while surfing the web during a given time frame. The author of [4] used three approaches for defining session. The first approach is IP and cookies, the next approach is IP, cookies and a temporal limit on intra session interactions and the last approach is IP, cookies and query reformulation patterns. The first and second methods are system oriented variables and the third method is the combination of user-oriented variables and system-oriented variables. For the first method they used the IP and cookies to identify unique users, the second method used a threshold of 30 minutes which was used to define two distinctive sessions of the same unique user. For the last method, the IP and cookies were used to identify unique users and the query reformulation pattern to identify the limit between two sessions considered as determined by a query which no term in common with the precedent one.

The most common and simplest method for the session identification is the threshold based mechanism, which is also known as the timeout method. In this method, a session is identified between two requests if the inter-arrival time between the two requests is greater than the predefined threshold value. The author of [5] used the threshold value of 100 sec, while the author of [6] used the threshold value of 1 sec. The result of this method is affected with different threshold value. This method is works well if the threshold value is correctly matched to the values of connections and session inter-arrival times. In [7], the authors used the timeout method on two web logs. Initially they set large threshold value and then gradually decreased. They concluded that the optimal threshold was found in the range of 10-15 minutes.

Other authors used different types of methods to identify web user session. The author of [3] developed an algorithm for session identification. They used page threshold with Frame page to develop the algorithm for identifying web user-session. They first identify specific users and then filter the frame page. The next and last step is to combine the contents of each page and all web structure forming actual session. In [8], the author used IP address, browsing agent, intersession and intra-session timeouts, immediate link analysis between referred pages and backward reference analysis to develop a technique for identifying user-session boundaries. They analyze with different server's logs and show that their technique identifies user session boundaries and generate all relevant user session sequences. The author of [10] develops an algorithm for session identification by using statistical language modeling.

## II. CLUSTERING TECHNIQUES

This section describes the overview of the clustering techniques. Informally, clustering can be defined as the process of dividing the given data into groups of similar objects in such a way that objects in one cluster are similar to each other and different from those objects which are in the other clusters. Clustering techniques are described in detail in [9]. The goal of this paper is to exploit this property of clustering to group connections to identify web user sessions.

Let $\boldsymbol{X}$ denotes the metric space, a set containing notions of metric or distance. This metric space is also known as sampling space. Let a set of samples which have to be grouped to form $\boldsymbol{K}$ cluster be

$$\boldsymbol{S=\{x_1, x_2, x_3, \ldots\ldots\ldots\ldots, x_N\}}$$

Where $\boldsymbol{x_i}$ belong to $\boldsymbol{X}$; $\boldsymbol{i}=1, 2, 3\ldots\ldots\ldots\boldsymbol{N}$.

The main purpose of this work is to find the partition $\boldsymbol{C= \{C_1, C_2, C_3, \ldots\ldots\ldots\ldots, C_K\}}$ such that $\boldsymbol{UiCi=S}$ and $\boldsymbol{Ci \cap Cj=\Phi}$. $\boldsymbol{C_1, C_2, C_3, \ldots\ldots\ldots\ldots, C_K}$ are clusters. Clusters contain similar samples while samples in the different clusters should be dissimilar. The measurement of similarity between two samples or two clusters plays an important role in clustering method since every clustering techniques depends on the similarity or dissimilarity between two data points. Clustering methods are of no used if there is no measure of similarity or dissimilarity between two data points.

To determine the dissimilarity between two objects, it is common to used distance measure. The Euclidean distance is commonly used for distance measure which is given by:

$$d_{euc}(x, y) = [\sum_{j=1}^{d} (x_j - y_j)^2]^{\frac{1}{2}}$$

### A. The Hierarchical Agglomerative Approach

The hierarchical agglomerative methods are based on measures of distance between clusters. This method merges those two clusters that are nearest to form a reduced number of clusters. This is repeated, each time merging the two closest clusters, until just one cluster, of all the data points, exists. Before the procedure starts, each sample is associated to a different cluster, i.e., $C_i=\{x_i\}$, thus the number of cluster $N_c$ is equal toN=|S|. Then, the nearest clusters are merged. This process is repeated and the procedure ends when

all the given samples are merged to the same cluster i.e. $C=S=\{x_1, x_2,\ldots\ldots..,x_N\}$ and $N_c=1$. At each step, the quality indicator function $\gamma^{(s)}$ is evaluated. Finally, the set S is clustered by using the optimal number of clusters $N_c=N-(i-1)$ such that $\gamma^{(i)}- \gamma^{(i-1)}$ is maximized. The quality indicator function measures the distance between the two closest clusters at step i. A sharp increase in the value of quality indicator function indicates the merging of two clusters which are too far apart.

B. The Partitional Approach

*k*-means algorithm is an algorithm of a partitioning clustering technique. In this algorithm, the number of clusters is fixed. This algorithm starts by selecting an initial partition with ***k*** clusters containing randomly chosen samples. Then we compute the centroids of the clusters. By generating a new partition, we assign each sample to the closest cluster center. Then we calculate new cluster centers as the centroids of the clusters. Centroid is defined as the mean value of the cluster samples.

i.e $$\hat{c}_i^k = \frac{1}{|C_i|}\sum_{x\in C_i} x^k \qquad k=1,2,\ldots\ldots\ldots..,n.$$

Where *n* is the size of the sampling space.

This process is repeated until the number of samples which are moved to different cluster is negligible.

## III. THRESHOLD BASED ALGORITHM

Threshold based algorithm is the simplest method that was used by many researchers for the identification of user-session. This algorithm is also known as the timeout method. The threshold based algorithm required to select a threshold value before running the algorithm. Based on this value, the algorithm identifies the sessions. So, the result of this algorithm is different for different threshold value i.e. threshold based algorithm depends on proper selection of threshold value. Threshold based algorithm gives accurate session if the pre-defined threshold value is correctly set. That's mean that optimal threshold value gives almost accurate session.

The algorithm works as follows:

The procedure starts with the proper selection of threshold value. Based on this value, the following condition is checked:

- If the inter-arrival time between the consecutive connections is less than the threshold value, then the two connections are in the same session.
- If the inter-arrival time between the consecutive connections is greater than the threshold value, then the two connections are in different sessions i.e. the first connection is the last element of the current session and the second connection is the first element of the next session.

The main limitation of this algorithm is that it requires a priori definition of the threshold value which is very difficult to set correctly. If the threshold value is too small, then the condition for separating two related connections in different sessions may occur. On the other hand, if the threshold value is too large then the condition for merging two unrelated connections in the same session may arrive. So, this algorithm is difficult to use in practice.

## IV. USING CLUSTERING TECHNIQUES ON WEB SERVER LOG

The log file used in our experiments was extracted from NASA website that can be freely downloaded from the link http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html. Before running the algorithm, the first step is to perform data cleaning for removing the irrelevant and redundant log entries. There are three main kinds of irrelevant information to be removed. First, embedded file which embedded in the web page should remove. A request made by a user to view a particular page usually results in several log entries because some embedded file such as graphics and scripts are downloaded along with the web page. It is unnecessary to include the file requests that the user did not explicitly request. Elimination of such information can be done by checking the suffix of the URL name. So, those entries which have the type extensions: .GIF, .JPEG and .JPG etc. are removed. The second irrelevant information to be removed is error's request. This can be removed by checking the status of request. There are four classes of status code. They are success (200 series), redirect (300 series), failure (400 series) and state error (500 series). Those entries that correspond to the status code which is not equal to 200, are removed. The last irrelevant information to be removed is the entries with the 'HEAD' method. The next step is to identify users. For user identification, we consider each IP address as a single user and we select those users which have more than 800 TCP connections for our experiment.

After user identification, the next step is to select a proper clustering technique for session identification. Hierarchical clustering technique is easy to implement but this technique does not scale well with large number of samples. Whereas partitional clustering technique require a priori knowledge of the number of clusters in advance which is not easy to predict but this technique is relatively efficient. In order to take the advantages and to avoid the limitations of both methods, we used both algorithms i.e. hierarchical and partitional algorithms. First partitional algorithm is used to obtain an initial clustering. This is performed by selecting a large number of clusters. But the number of clusters must be less than the number of samples. The next step is to apply hierarchical agglomerative clustering algorithm to the result obtained from first step. The hierarchical agglomerative clustering technique is used to combine the

clusters in order to obtain a good estimation of the final number of clusters. To get a fine definition of number of clusters, a partitional clustering algorithm is used as a last step. The detail description of each step is given below.

A. Partitional clustering approach

The first step starts with *k*-mean clustering method with ***k*** cluster. The ***k*** cluster is chosen as large as possible but it should be less than the number of samples. The distance between any two adjacent samples is then calculated. Farthest ***(k-1)*** couples are taken according to the distance metric to determine ***k*** intervals. Let $T_{int}$ be the inferior bounds of the intervals and $T_{sup}$ be the superior bound of the intervals. Then the centroid of each cluster is calculated as

$$centroid=( T_{sup} + T_{int})/2$$

The partitional algorithm is then iteratively run to obtained ***k*** initial clusters. Each cluster ***C*** is represented by a small subset ***R(C)*** of samples. **|*R(C)*|** may be less than or equal to *2* since the metric space is ***R***. The representative samples ***R(C)*** may be either the cluster centroid or single linkage procedure.

B. Hierarchical agglomerative approach

A hierarchical agglomerative algorithm is iteratively run using representative samples obtained from step 1. The number of steps is ***k*** since the procedure starts with ***k*** initial clusters. Hierarchical agglomerative procedure merges the two closest clusters at each step. After merging the two clusters, distances between clusters are recomputed. This process is repeated. The process ends after ***k*** iterations.

The clustering quality indicator function which measures the distance between the two closest clusters is used to select the best clustering among those determined in the iterative process. Quality indicator function is denoted by $\gamma^{(s)}$**.** At each step ***s***, the clustering quality is calculated to find if the optimal number of clusters has been found. Let $C_j^{(s)}$ denote the $j^{th}$ cluster of step ***s*.** At each step, the procedure evaluates the quality indicator function $\gamma^{(s)}$:

$$\gamma^{(s)} = \frac{d_{\min}^{(s)} - d_{\min}^{-(s)}}{d_{\min}^{-(s)}}$$

Where

$$d_{\min}^{(s)} = \min_{j,k \neq j} d(C_j^{(s)}, C_k^{(s)}),$$

$$d_{\min}^{-(s)} = \frac{1}{s-1} \sum_{l=1}^{s-1} d_{\min}^{(l)},$$

A sharp increase in the value of $\gamma^{(s)}$ indicates that the merging procedure is artificially merging two clusters which are too far apart.

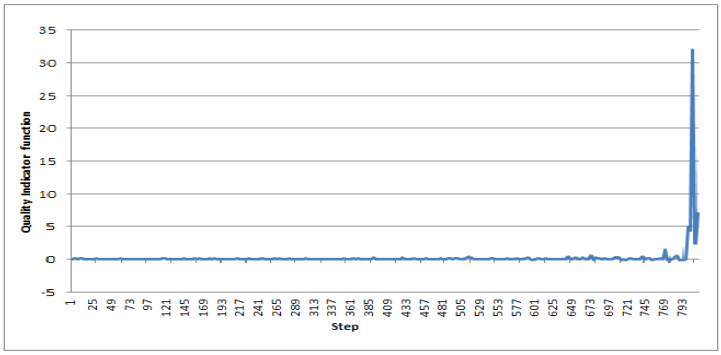


Figure 1: Sample plot of the quality indicator function.

The optimal number of clusters $N_c$ is determined as:

$$N_c=N-(argmax_s(\gamma^{(s)}- \gamma^{(s-1)})-1)$$

C. Final clustering creation

The last step is to find a fine definition of $N_c$ clusters. In this step, a partitional clustering algorithm is run over the original data which contain all samples to get a final refinement of the clustering definition.

## V. PERFORMANCE ANALYSIS

In this section, the correctness properties of the clustering scheme are described and compared it with that of the threshold-based mechanism. The percentage of misidentified sessions is used as the performance metric. Two types of errors may occur: (i) to erroneously separate in two or more clusters or (ii) to merge two or more distinct sessions. The percentage of errors is defined as 100 times the total number of observed errors divided by the total number of connection arrival times.

Proper choice of threshold value gives almost real session [7]. We consider this value as optimal threshold value. If the clustering algorithm is more accurate, then its graph (i.e. T_off- error percentage graph) must be parallel to x-axis when we assume x-axis as the optimal threshold graph.

In this section, we first determine the optimal threshold value from the web server log. Then, the threshold based algorithm is run using optimal threshold value. After that, threshold based algorithm is again run with different threshold value and the results are compared with that of the sessions that is obtained by using optimal threshold value in order to get the misidentified session. Similarly, clustering algorithm is run to identify sessions and compared with that of the sessions obtained by using optimal threshold value to get the misidentified sessions. By using these misidentified sessions as the performance metric, we compared the clustering algorithm and threshold based algorithm.

### A. Determination of optimal threshold value:

We used iteration number of a session as the number of TCP connections in a session. For example, if the iteration number of a session is five, then the session has five connections. A sequence of connections is grouped into a session if and only if

- The connections are from the same user ID or IP address
- The time interval between two adjacent connections is less than or equal to the session interval in use.

By grouping the sessions with the same iteration number, we can see the distribution of various sessions. The distributions show the percentages of sessions with a particular iteration in relation to the total number of sessions. This was done because different session intervals cut the logs into different number of sessions, and so a percentage comparison was more meaningful.

Our experiment and analysis focused on monitoring only the distributions of the session with less than or equal to 6 iterations because their total covers a very large percentage of sessions.

Table I: The results of session interval from the web logs.

| Session interval (in sec) | Percentage of 1 iteration | Percentage of 2 iteration | Percentage of 3 iteration | Percentage of 4 iteration | Percentage of 5 iteration | Percentage of 6 iteration |
|---|---|---|---|---|---|---|
| 60 | 80% | 13% | 3% | 1% | 0% | 0% |
| 80 | 72% | 16% | 5% | 1% | 0% | 0% |
| 100 | 71% | 17% | 6% | 2% | 1% | 0% |
| 120 | 68% | 18% | 6% | 2% | 1% | 0% |
| 140 | 65% | 18% | 8% | 2% | 1% | 1% |
| 160 | 64% | 18% | 8% | 3% | 1% | 1% |
| 180 | 62% | 19% | 8% | 3% | 1% | 1% |
| 200 | 61% | 19% | 9% | 3% | 1% | 1% |
| 220 | 59% | 19% | 9% | 4% | 2% | 1% |
| 240 | 57% | 19% | 10% | 4% | 2% | 1% |
| 260 | 56% | 19% | 10% | 4% | 2% | 1% |
| 280 | 55% | 20% | 10% | 5% | 2% | 1% |
| 300 | 54% | 20% | 10% | 5% | 2% | 1% |

We took 60 to 300 seconds as a range of session interval. After 300 second, graphs of all iterations are almost stable. More importance is given to the graph with smaller number of connections in a session while observing optimal session interval. Sudden dramatically

change indicates the existence of optimal session interval. Iteration graph increase gradually from 300 second to 80 second. From 80 second, it increases dramatically. Similarly other graphs shows stable from 300 second to 80 second, after 80 second, it shows sudden decrease.

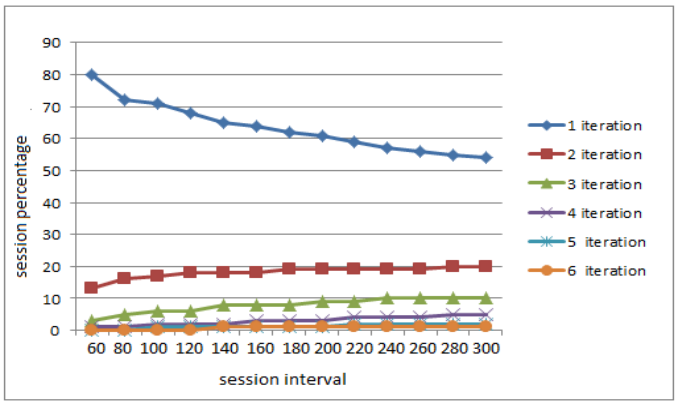


Figure 2: The result of session interval from the web logs.

The optimal threshold value should not be too large nor too small. The results of the experiment show that most sessions were not affected when the session interval is larger than 80 seconds. When the session interval is less than 80 seconds, most sessions are affected. When the session interval becomes smaller, the percentage of sessions with 1 iteration increases dramatically whereas the percentages of sessions with 3-6 iterations, decreases dramatically. This shows that the optimal session interval is nearly 80 seconds. Hence, we take the optimal threshold value to be 80 seconds.

B. Parameter sensitivity:

In our clustering algorithm, initial k value (i.e. number of clusters) is assigned in order to run the algorithm. But its value is replaced when we run the final partitional algorithm with the value from hierarchical agglomerative algorithm. Therefore, k-value must be independent of its selection.

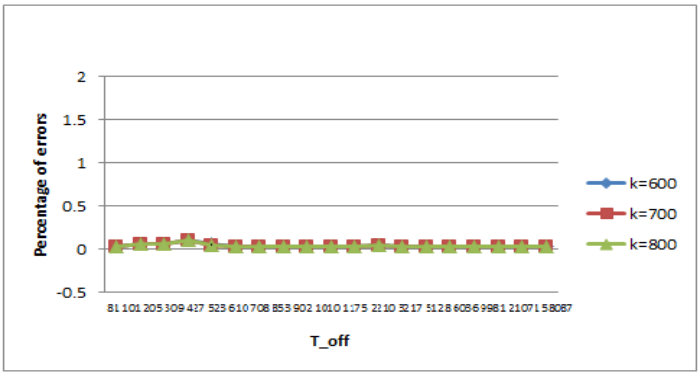


Figure 3: Clustering sensitivity to the initial number of clusters k.

Figure 3 shows the error probability of clustering algorithm by taking initial number of cluster, k=600, 700 and 800. This graph shows that error probability is independent from the value of k, since graphs are superimposed. Hence, initial number of cluster is not a critical parameter.

C. Percentage of misidentified sessions:

We consider the percentage of sessions misidentified by the clustering procedure to assess the quality of the results. The result thus obtained is compared with that of the threshold based mechanism.

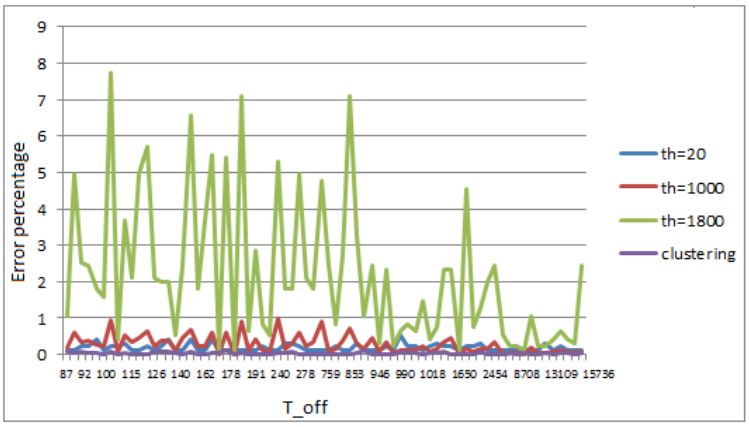


Figure 4: Percentage of errors obtained by running clustering and threshold based mechanism.

The above figure shows the percentage of errors obtained by running threshold based scheme and clustering scheme. Performance of the threshold based mechanism is evaluated by taking threshold value of 20, 1000 and 1800 seconds. From the above figure, we see that clustering graph is almost parallel to x-axis which shows that it is less error than other threshold algorithms. Threshold algorithms (except optimal threshold) give irregular graph to the x-axis.

From this, we conclude that threshold based mechanism may not perform well if the threshold value is not correctly set. Its error probability is much larger than the clustering scheme. When the value of threshold is too large, the percentage of errors is high which is due to the result of merging two subsequent different sessions. Small value of threshold value also induces high percentage of error which is due to the result that $T_{off}$ goes below a given value. Hence, the percentage of errors for clustering scheme is always less than 0.5% and it is less sensitive to the variation of $T_{off}$.

## VI. CONCLUSION

Clustering techniques are used for inferring web user-session. In order to identify the web user session, a combination of the hierarchical and partitional clustering techniques is used. First, a *k*-mean partitional algorithm is used to obtain an initial clustering. After getting the initial clustering, a hierarchical agglomerative clustering technique is run to the representative samples obtained from the first step to obtain the final number of clusters. Finally, partitional clustering is applied to the real data to obtain the fine definition of clusters. This process for identification of web user-session can deal with large amount of data. The effectiveness and robustness of the clustering techniques was used to show its ability in the identification of web user-sessions without requiring any a priori definition of threshold values. This may be used in characterizing web user-sessions and in the study of internet traffic properties.

There are some limitations which could be directions for future work. First of all, the resulting algorithm is applied only to the web traffic. So, this work can be extended in order to apply to different types of traffic and to deal with other types of user sessions which are not related to the web. After the identification of user-session, second identification is required to remove errors in the session pages and unrelated pages so that it can improve the quality of identified session.

## ACKNOWLEDGMENT

Many thanks to the Head Of Department, faculty members of Department of Computer Science and Engineering of JIET college for their motivation and constant encouragement. The authors would like to thank the family members and friends who rendered their support throughout this research work.

## REFERENCES

[1] Mat-Hassan M. and Levene M , “Associating search and navigation behavior through log analysis”, Journal of the American society for information science and technology, vol. 56(9), pp. 913-934, 2005.

[2] Bianco, A. Mardente, G. Mellia, M. Munafo, M. and Muscariello, L., "Web User-Session Inference by Means of Clustering Techniques," Networking, IEEE/ACM Transactions on , vol.17, no.2, pp.405-416, April 2009.

[3] Fang Yuankang and Huang Zhiqui, “A session identification algorithm based on frame page and page threshold”, Computer Science and Information Technology (ICCSIT), 3rd IEEE International Conference, 2010.

[4] Jansen B. , Spink A. , Blakely C. and Koshman S., “Defining a session on web search engines”, Journal of the American society for information science and technology, vol. 58(6), pp.862-8871, 2007.

[5] C. Nuzman, I. Saniee, W. Sweldens, and A. Weiss, “A compound model for TCP connection arrivals, with applications to LAN and WAN,” *Computer Networks, Special Issue on Long-Range Dependent Traffic*, vol. 40, no. 3, pp. 319–337, Oct. 2002.

[6] F. D. Smith, F. H. Campos, K. Jeffay, and D. Ott, "What TCP/IP protocol headers can tell us about the web," *SIGMETRICS Perform. Eval. Rev.*, vol. 29, no. 1, pp. 245–256, 2001.

[7] He D. and Goker A., "Detecting session boundaries from web user logs", In proceedings of the BCS-IRSG 22nd annual colloquium on information retrieval research, Cambridge, UK, pp. 57-66, 2000.

[8] Arumugam G. and Suguna S.,"Optimal algorithms for generation of user session sequences using server side web user logs", Network and service security, International Conference on 24-26 June 2009.

[9] Guojun Gan, Chaoqun Ma, Jianhong Wu, "Data Clustering: Theory, Algorithm and Applications", ISBN: 0898716233, Society for Industrial and Applied Mathematics (SAM), 2007.

[10] Xiangji Huang, Fuchun Peng, Aijun An, Dale Schuurmans, "Dynamic Web Log Session Identification With Statistical Language Models", Journal of the American Society for Information Science and Technology, 2004.

## AUTHORS

**First Author**- Nirmala Huidrom, M.Tech student, Department of Computer Science and Engineering, Jodhpur Institute of Engineering and Technology, Jodhpur, India.
e-mail: nirmala_huidrom@yahoo.co.in

**Second Author**- Neha Bagoria, Asst. Professor, Department of Computer Science and Engineering, Jodhpur Institute of Engineering and Technology, Jodhpur, India.