Semantic Based Weighted Web Session Clustering Using Adapted K-Means and Hierarchical Agglomerative Algorithms

Sowmya HK* and R. J. Anandhi

Department of Information Science and Engineering, New Horizon College of Engineering, Affiliated to Visvesvaraya Technological University, Bengaluru, India
E-mail: sowmyahk2020@gmail.com; rjanandhi@hotmail.com
**Corresponding Author*

Received 02 March 2021; Accepted 15 October 2021;
Publication 03 January 2022

Abstract

The WWW has a big number of pages and URLs that supply the user with a great amount of content. In an intensifying epoch of information, analysing users browsing behaviour is a significant affair. Web usage mining techniques are applied to the web server log to analyse the user behaviour. Identification of user sessions is one of the key and demanding tasks in the pre-processing stage of web usage mining. This paper emphasizes on two important fallouts with the approaches used in the existing session identification methods such as Time based and Referrer based sessionization. The first is dealing with comparing of current request's referrer field with the URL of previous request. The second is dealing with session creation, new sessions are created or comes in to one session due to threshold value of page stay time and session time. So, authors developed enhanced semantic distance based session identification algorithm that tackles above mentioned issues of traditional session identification methods. The enhanced semantic based method has an accuracy of 84 percent, which is higher than the Time based

Journal of Web Engineering, Vol. 21_2, 239–264.
doi: 10.13052/jwe1540-9589.2125

and Time-Referrer based session identification approaches. The authors also used adapted K-Means and Hierarchical Agglomerative clustering algorithms to improve the prediction of user browsing patterns. Clusters were found using a weighted dissimilarity matrix, which is calculated using two key parameters: page weight and session weight. The Dunn Index and Davies-Bouldin Index are then used to evaluate the clusters. Experimental results shows that more pure and accurate session clusters are formed when adapted clustering algorithms are applied on the weighted sessions rather than the session obtained from traditional sessionization algorithms. Accuracy of the semantic session cluster is higher compared with the cluster of sessions obtained using traditional sessionization.

Keywords: Sessionization, dissimilarity matrix, session weight, session cluster, cluster evaluation.

1 Introduction

Web mining is the process which uses techniques and algorithms of data mining to extract knowledge from the Web. In order to provide insight into the growth of the industry and users, the primary goal of web mining is to acquire web data trends by analysing information. There are three types of web mining, such as web structure mining, web content mining and web usage mining [14]. Web content mining discovers knowledge from conventional collections of multimedia documents, such as images, video, and audio, implanted in or connected to web pages. To find more suitable documents by analysing the connection structure of the web, Web Structure Mining uses graph theory to examine the node and connection creation of a web site. Web Usage Mining focuses on approaches that can predict behavioural intention when communicating with the WWW. It uncovers visitor navigation trends and attempts to discover the web log file information that is relevant.

Web log files are an archive of user activities documented while visiting a website. The web server stores these log files and are available in different formats like Common Log Format and Extended Log format. Web log files which include attributes such as user name, IP address, request time stamp, number of bytes transmitted, user agent and referrer. Investigation of these log files reveals browsing operation of the user. But these files are very huge and does not provide clear sketch of user's request to the website. Data pre-processing is also one of the most significant steps in the mining of web use to discover information.

There are three significant stages of web usage mining, such as data pre-processing, pattern discovery, and pattern analysis. Data pre-processing is used to clean data from web logs in order to make it suitable for pattern discovery. Data pre-processing can be done in several steps like data fusion, data extraction, data cleaning, session identification and user identification. With the assistance of data mining techniques such as association rule mining, clustering and classification, pattern discovery can be achieved. Pattern analysis is used to eliminate irrelevant rules and patterns and can be fed as input to applications such as visualization tools and generation tools.

The session is described as a collection of pages that the user has accessed during his visit to the website. During their visit to the web site, a user may have one or more sessions. Session identification is major and sophisticated task in web usage mining because, pages accessed by a user aren't found in relevant order within the server log. There could be sizable amount of users visiting the web site, so their requests are going to be entered within the log simultaneously, which creates jumbled entry within the log. As a result, session identification techniques are critical for detecting legitimate online user sessions.

Traditional session identification techniques, such as time-based and referrer-based methods, have their own set of constraints. Because these two algorithms break a single session into many sessions depending on time and referrer information, they yield denser sessions. The goal of this study is to derive more accurate sessions by considering all three aspects of time, referrer, and semantics. This study defines an effective approach for constructing user session which is based on the semantic difference between the URLs visited by the user, time and referrer information.

The grouping of user sessions based on browsing activity is also discussed in this study. The knowledge gained via clustering can be used to assess the user's pattern of web site usage, advocate web site restructure, pre-fetch or cache pages, and forecast the next page viewed by the user to minimize latency and so on. The researchers calculated the web page's page weight based on how much time they spent on it, which was then used to calculate session weights. The dissimilarity matrix, which is an input for the clustering algorithm, is created using these two parameters. Clustering algorithms such as adapted Hierarchical agglomerative and K-Means are applied on weighted sessions for clustering similar web sessions. This helps to draw inference about usage of web.

The rest of the paper is organised as follows. A literature review of existing session identification and clustering algorithms for web usage mining

is discussed in Section 2 of the article. The proposed semantic dissimilarity-based session identification is provided in Section 3. The page weight and session weights are calculated in Section 4 based on the user's browsing activity. The methodology for estimating the dissimilarity matrix is provided in Section 5. In Section 6, we look at how to cluster weighted web sessions using customised K-means and agglomerative algorithms. The Dunn Index and Davies-Bouldin are used to evaluate clustering techniques in Section 7. The experimental study is carried out in the penultimate portion. The last section discusses the conclusion.

2 Literature Survey

The study of a web log file is a difficult task since it includes a vast amount of information about a web page. Depending on the network layout, URLs are parsed as tokens. The authors have introduced a new approach for clustering web URLs based on the sequence of pages visited by the user [1]. Members from each cluster may be defined since it contains the largest number of potential pages in the sequence. It also assists in the pre-fetching of webpages for user searches. A web page recommendation framework [6] provided users with recommendations by taking into account the order of events that occurs in website page usage trends. Here, for each page, authors assessed page weight and projected top page suggestions for each target user. Users close to target users have been identified using the fuzzy C-mean clustering algorithm.

A modified K-means clustering algorithm [7] is designed for grouping web user sessions. Authors provided an approach to determine the sessions distance based on their web access trace affinity that took care of variable duration length of user sessions. This paper did not address the impression of their technique on more number of session and cluster. A method for analyzing web logs by mining Directed Acyclic Graph access patterns based on the browsing time of each page was proposed by Mihara et al. [5]. They also proposed a method for pattern reduction performing closed pattern mining and clustering of similar patterns. These patterns are helpful for the website admin to figure out issues of their web sites in more efficient manner and to set up solid plans to enhance their sites more easily.

Kaur et al. [10], presented semantic-time-referrer based algorithm for session identification from web log file. This method enhances the attainment compare to traditional session identification algorithm by suggesting time, referrer and semantic concept. There is advancement in both credibility and productivity of the algorithm. It calculates the semantic difference between the previous URL request and the current one, even if the field of reference

is empty. Time component added to this algorithm avoids formation of long session. But authors have not used generated session for extracting usage pattern.

When a user interacts with a website, his acts are solely decided by himself and other external influences such as the structure and dynamicity of the website. These considerations are taken into account when determining user behaviour [25]. To improve the modelling of dynamic user behaviour, a mixture of short and long term modelling techniques is used. This is important for developing web services. A novel user identification algorithm built on MapReduce technique is suggested by Srivastava et al. [4]. Here the IP address and knowledge of the user agent are used for user identification. The presented method handles various issues such as robot detection and scalability. Authors have not taken into account information such as referrer and website topology for recognizing users.

The fuzzy method for applying weights to user sessions is suggested by Ansari et al. [11]. The suggested clustering algorithm overcomes the challenge of selecting initial cluster centres. Experimental results demonstrated the quality of clusters obtained and clustering algorithm's performance in terms of time. To boost the correctness and precision of the clustering algorithm, Lu proposed K-means clustering under the Hadoop parallel framework [23] in order to improve agglomeration accuracy, the author presented the weighted K-means and weighted fusion K-means rule that diminishes the effect of error metrics on the accuracy of the agglomeration. This experiment, however, only uses four computers due to the restriction of experimental constraints, scarcity of massive cluster examination, and testing of greater information sets involves further validation.

A detailed study is performed in order to understand their connectivity in the sense of large data sets, different clustering methods were analyzed from a hypothetical perspective and scientifically evaluated on artificial benchmarks to illustrate their benefits and limitations [12]. Within the benchmark data sets, the algorithms could not create clusters and had downfalls such as tolerance to noise and anomalies, complexity of time and process, and inability to find clusters. A two-phase novel MapReduce based hybrid cluster algorithmic rule for decentralized sets of data was presented by Sinha et al. [24]. It's been shown that the solution uses the genetic rule options needed, Mahalanobis distance and k-means++. The experimental findings are that mrk-means, P k-means and scaling GA are surpassed by the expected algorithmic program. The proposed rule, however, only applies to static datasets.

The following is a summary of major existing works on pre-processing and clustering.

Table 1 Summary of key existing works on pre-processing and clustering

Author	Methodology	Remarks
G. Poornalatha et al. [7], 2011	The distance between any two sessions is computed using the Variable Length Vector Distance approach in the modified K-Means clustering technique	The authors looked at the frequency of web page categories and used the Jaccord coefficient to determine how similar the clusters were. Algorithm efficiency is not measured for varying number of clusters.
M. Srivastava et al. [4], 2018	The authors presented a Map-Reduced based methodology for user identification that employs IP address and user agent as parameters.	The proposed technology offered a solution to problems including proxy servers, robot detection, and scalability. However this work is only limited to the identification of users.
Kaur et al. [10], 2017	To compute the web user session, time-referrer based and semantic-time-referrer based algorithms were presented.	In comparison to traditional algorithms, better sessions are generated. The scope of this work, however, is confined to identifying user sessions.
Z. A. Ansari et al. [11], 2016	To find and validate web session clusters, the proposed technique use the Mountain Density Function.	This work suggested a method for giving fuzzy weights to user sessions and associated URLs that takes into account the large dimensionality of the user session.
W. Lu [23], 2020	To work on enormous datasets, the author created a parallel K-means clustering technique based on density. To limit the impact of incorrect parameters, a weighted k-means algorithm is utilised.	With the support of a cloud platform, the proposed strategy was able to attain a higher acceleration ratio when dealing with massive datasets. But, the quality of the cluster is not measured in this work.
Proposed Method	The proposed technique provides a precise way for calculating semantic differences between URLs, which is employed in the session identification algorithm together with time and referrer information. It also proposes a modified agglomerative and K-means algorithm to cluster web user sessions based on their browsing habits.	This paper outlines a method for establishing high-quality sessions while avoiding the creation of bogus sessions. Based on the user's browsing behaviour, it calculates page weight and session weight, which are then utilised to generate more accurate semantic clusters. The Dunn Index and Daves-Bouldin Index are used to assess cluster quality.

3 Proposed System

Present work proposes a novel session identification algorithm to obtain user sessions from web log files. It forms web session, based on important criteria such as time, referrer and semantic distance between the URLs. Session weights are computed based on the weights assigned to each visited page which belongs to that session. Proximity matrix is estimated based on session weights. Then adapted Hierarchical clustering algorithm and K-means clustering algorithm is applied on this proximity matrix to discover cluster of similar web sessions based on remarkable web pages from web log file.

When the user visits few pages more frequently and spend more time shows the importance of those pages. Visitors have visited these pages multiple times and taken more time because they carry more applicable information than other. In the traditional system, even if the page stay time is limited, the value of the page is measured solely by the amount of times it is viewed.

The session includes a series of URLs that the user navigates. Pages navigated by a visitor are identified according to each URL in the session. Proposed system is intended to obtain clustering of web sessions. Depends on the weight of web pages that are part of the session, sessions are clustered. Grouping of sessions based on weighted pages will result in more significant clusters. Figure 1 show the process involved in proposed weighted session clustering.

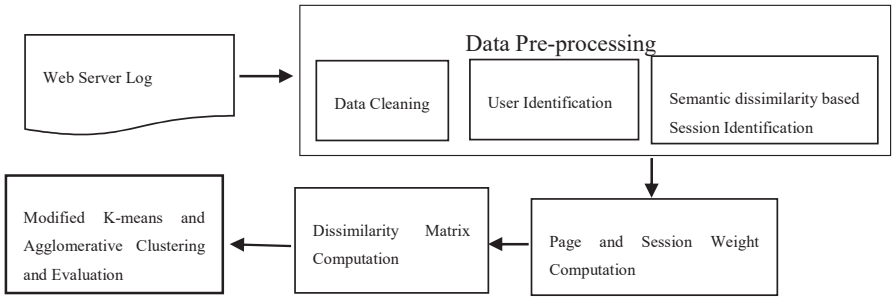

Figure 1 System architecture.

3.1 Data Pre-Processing

Data Pre-processing is a significant step of the mining activity of web usage. It consists of set of activities such as data cleaning, user identification, session identification.

3.2 Data Cleaning

Data cleaning is the process of removing irrelevant and incomplete information from a web log file. It also eliminates log entries with a failing status code, as well as access to image files like jpeg, GIFF, and others. When a user accesses a web page, it may contain system generated advertisements in different formats which makes mining task complex. Remove all such unwanted details from the web log and retain only the details what user has requested. Data cleaning process improves the quality of data and hence it improves accuracy of pattern discovery and analysis [17, 18].

3.3 User Identification

User identification [20] activity helps to identify unique users visited that web site. It becomes complex task when the user ID field is missing in the web log file. It is difficult to process because, one user may use several IP address or several user may use same IP address using proxy server or user may visit a site more than once. So some heuristics need to be identified to recognize unique user from web log file.

3.4 Session Identification

Session is a collection of pages that a user visits during his or her single web visit. Session identification algorithm divides the web page access entry from cleaned web log in to individual sessions. The conventional time-oriented sessionization method [21] uses only time limits for session duration and page stay time to build sessions. The conventional referrer-based method [10] generates a session using only the log fie referrer field.

Time-based Session Identification

The time-oriented session identification method is a well-known method that takes into account time limits, such as a maximum session length of 30 minutes or a maximum Page view time of 10 minutes. When the Page stay time or session duration time exceeds the threshold value, a new session is

created, according to this approach. If the time gap is greater than 10 minutes, a new session is created; however, if the navigation path is inspected, the session should keep the same. As a result, the most significant restriction of this strategy is that most of the time, a single session is broken into multiple sessions or numerous sessions are combined into a single session.

Time-referrer Based Session Identification

The traditional referrer-based technique creates sessions using the log file's referrer field. This method compares the referrer field of the current URL request to the referrer field of the most recent URL request. To achieve better results, the referrer – time oriented technique has integrated the time based and referral based approaches, as well as some additional tests.

Semantic Dissimilarity Based Session Identification Algorithm

Proposed session identification method combines both time and referrer based methods with additional semantic check to yield better quality of sessions.

The semantic difference is calculated using the following equation between two URLs.

$$\mathrm{Sdiff}(\mathrm{Url}_1,\mathrm{Url}_2) = \frac{Max\_Count(Url1, Url2) - Match\_Count(Url1, Url2)}{Max\_Count(Url1, Url2)} \tag{1}$$

The variable Sdiff, stores result of the above computation. The function Match_Count(Url_1,Url_2), finds the number of web page matches among Url_1 and Url_2. It tries to match from left to right, if there is any mismatch, it stops counting. The function Max_Count(Url_1,Url_2) returns, maximum number of web pages present in either Url1 or Url2.

For example, if the Url_1 = "Home/Admin/Allotment" and Url_2 = "Home/Admin/selection", the number of common web page between two URL is 2. The maximum count of web page among Url_1 and Url_2 is 3. So the result will be $(3-2)/3 = 0.33$.

Algorithm:

Input: Visitor = $\{\mathrm{v}_1,\mathrm{v}_2,\mathrm{v}_3,\ldots \mathrm{v}_n\}$, Threshold = 0.3, $\mathrm{Threshold}_{\mathrm{SessionTime}}$ = 30 Minutes, $\mathrm{Threshold}_{\mathrm{Pstaytime}}$ = 10 Minutes
Output: Session = $\{\mathrm{v}_1 = (\mathrm{s}_1,\mathrm{s}_2,\mathrm{s}_3\ldots \mathrm{s}_\mathrm{m}),\ \mathrm{v}_2 = (\mathrm{s}_1,\mathrm{s}_2,\mathrm{s}_3\ldots \mathrm{s}_\mathrm{m}),\ldots \mathrm{v}_\mathrm{n} = (\mathrm{s}_1,\mathrm{s}_2,\mathrm{s}_3\ldots \mathrm{s}_\mathrm{m})\}$

Session_Count=0
for each visitor from v_1 to v_n
  for each request of visitor v_i
if(referrer==NULL)
if(Sdiff(Cur_URL,Prev_URL)<=Threshold)
  Continue
else if((T_{cur_req}–T_{prev_req})<=$Threshold_{Pstaytime}$)
  Continue
else if((T_{cur_req}–T_{first_req})<=$Threshold_{SessionTime}$)
Continue
  else construct unique Session, Session_Count++
else
if(Referrer==Prev_URL)
  Continue
else if(Referrer==Prev_Referrer)
Continue
else if(Sdiff(Cur_Referrer,Prev_Referrer)<=Threshold)
Continue
else if(Sdiff(Cur_Referrer,Prev_URL)<=Threshold)
Continue
else if(Sdiff(Cur_URL,Prev_URL)<=Threshold)
  Continue
else if((T_{cur_req}–T_{prev_req})<=$Threshold_{Pstaytime}$)
  Continue
else if((T_{first_req}–T_{cur_req})<=$Threshold_{SessionTime}$)
  Continue
else construct Unique Session, Session_Count++

Here, the algorithm uses referrer field value of visitor request to check various criteria. It checks for three conditions if this field is NULL. It first detects the semantic distance between the current and previous URL, second compares the current and previous URL queries in terms of time, and third computes the difference in time between the current and previous URL queries. If a condition is met at any point, the algorithm goes on to the next entry log file. Otherwise, a new session is started. When the referrer field isn't NULL, it checks two referrer criteria. First, the current referrer field is compared to the previous URL request, and then the current referrer is compared to the previous referrer. If a condition is met at any point, the algorithm goes on to the next entry log file. Otherwise, a new session is started. When the referrer

field isn't NULL, it checks two referrer criteria. First, the current referrer field is compared to the previous URL request, and then the current referrer is compared to the previous referrer. To determine the semantic difference between current and previous referrer, current referrer and previous URL, and current and previous URL, three semantic checks are used. It checks time limits at the conclusion. If any of the conditions are met, the algorithm advances to the next log item. It creates new session only if, none of the conditions hold true.

4 Page and Session Weight Computation

Database is built from the pre-processed web log file with set of attributes such as IP address, Request date and time, page URL. Now compute weight for the visited page based on page stay (dwelling) time of visitor and access frequency. Thus dwelling time of URL is considered as one of the attribute for each entry in the web log. Dwelling time is computed by using following Equation (2).

$$\mathrm{DT}(\mathrm{P_i}) = \mathrm{t}(\mathrm{P_{i+1}}) - \mathrm{t}(\mathrm{P_i}) \tag{2}$$

Dwelling time of page $\mathrm{P_i}$ is calculated by subtracting requested time of $\mathrm{P_i}^{\mathrm{th}}$ page from requested page $\mathrm{t}(\mathrm{P_{i+1}})$ as given in Equation (2).

Table 2 shows sample users log representing page dwelling time of the respective pages in the web session. For example, it is assumed that in a day 6 visitors have browsed 6 pages in total. If user has visited a page and stayed in that page for minimum 1 minute, then entry is made in the table, otherwise it is considered as 0. If a user visits a page more than once in the same session, then dwelling time will be added up.

Table 2 Sample user's log of User ID = 1

	Page Stay Time (Min)					
Session ID	P_1	P_2	P_3	P_4	P_5	P_6
S_1	7	4	0	0	11	0
S_2	1	2	0	0	2	0
S_3	6	4	1	1	13	0
S_4	7	3	6	0	4	1
S_5	0	7	0	0	5	2
S_6	2	8	0	2	6	6

Let $\mathrm{P_1}$, $\mathrm{P_2}$, $\mathrm{P_3} \ldots \mathrm{P_n}$ be the collection of web page URLs in the web log file that is pre-processed. If a visitor hits a $\mathrm{P_1}$, $\mathrm{P_2}$, $\mathrm{P_5}$ page during 30 minutes, the session can be described as $\mathrm{S_1} = \{\mathrm{P_1}, \mathrm{P_{2,}}\ \mathrm{P_5}\}$

The weight of that session is portrayed as being

$$\mathrm{W}\ (\mathrm{S1}) = \{7, 4, 11\}$$

5 Dissimilarity Matrix (Proximity) Estimation Based on Session Weight

Traditional way of finding proximity matrix using three important message such as single linkage, complete linkage and average linkage [22]. Here, dissimilarity matrix for clustering web session is computed based on the distance between weighted sessions. It uses modified version of average linkage method and is derived by making use of following Equation (3).

$$\mathrm{D}(S_i, Sj) = \frac{1}{NP} \sum_{k=1}^{Np} |S_i(W(P_k)) - Sj(W(P_k))| \tag{3}$$

The above equation computes the distance between session $\mathrm{S_i}$ and $\mathrm{S_j}$. It computes sum of the differences between weight of visited pages of session $\mathrm{S_i}$ and $\mathrm{S_j}$ and then result is divided by total number of web pages visited by the user in that session.

Algorithm: Dissimilarity matrix between the sessions

Input: Set of web session with weighted pages
$\mathrm{S} = \{\mathrm{S_1}\text{–W }(\mathrm{S_1}), \mathrm{S_2}\text{–W }(\mathrm{S_2}), \mathrm{S_3}\text{–W }(\mathrm{S_3}) \ldots \mathrm{S_n}\text{–W }(\mathrm{S_n})\}$
Output: Similarity matrix D
Begin
For every User Session $\mathrm{S_i}$ to $\mathrm{S_n}$
For every User Session Sj to Sn
Compute D(i,j) $= \frac{1}{NP} \sum_{k=1}^{Np} |S_i(W(P_k)) - S_j(W(P_k))|$
End

Consider Table 2 data to illustrate the calculation of distance between two sessions. Now compute distance between session $\mathrm{S_1}$ and $\mathrm{S_2}$ as follows

$$\begin{aligned}\mathrm{D}(\mathrm{S_1}, \mathrm{S_2}) &= \frac{1}{NP} \sum_{k=1}^{Np} |S_1(W(P_k)) - S_2(W(P_k))| \\ &= \frac{1}{6}(|(7-1)| + |(4-2)| + |(0-0)| + |(0-0)|\end{aligned}$$

$$
\begin{aligned}
&+ |(11-2)| + |(0-0)|) \\
&= 2.8 \\
\mathrm{D}(\mathrm{S}_1, \mathrm{S}_3) &= \frac{1}{NP} \sum_{k=1}^{Np} |S_1(W(P_k)) - S_3(W(P_k))| \\
&= \frac{1}{6}(|(7-6)| + |(4-4)| + |(0-1)| + |(0-1)| \\
&+ |(11-13)| + |(0-0)|) \\
&= 0.8
\end{aligned}
$$

The above illustration shows that session S_1 and S_3 are more similar than session S_1 and S_2. Though session S_1 and S_2 are having similar browsing pattern, they are not so close because the user interest on those pages (page weight) are different.

6 Clustering of Weighted Web Session

Outcome of sessionization process is considered for clustering web session to analyse navigation pattern of the web site by the user. This paper proposes modified hierarchical agglomerative clustering approach to perform session clustering. Each session begins with specific cluster, and then the couple of clusters having the lowest dissimilarity value are considered and combined subsequently till all clusters aggregated into a unified cluster containing everything. Here, each session with navigation pattern as well as its consolidated derived weight of web page is considered as initial single cluster. A couple of combined clusters focused on the Average Linkage Feature (ALF). Using ALF, the distance between two clusters $\mathrm{C_i}$ and $\mathrm{C_j}$ can be calculated as

$$\mathrm{D}(\mathrm{C_i}, \mathrm{C_j}) = \frac{Tij}{Ni * Nj} \tag{4}$$

Here, $\mathrm{T_{ij}}$ denotes summation of pair wise data points distances between clusters i and cluster j. $\mathrm{N_i}$ and $\mathrm{N_j}$ represents cluster size of $\mathrm{C_i}$ and $\mathrm{C_j}$.

Algorithm: Adapted Hierarchical Agglomerative Clustering

Input: Dissimilarity matrix D, and total number of session n, Threshold
Output: Set of q Clusters $\mathrm{C} = \{\mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3 \ldots \mathrm{C}_q\}$
Begin

Count=n

```
while(Count>=1)
do
    for i: 1 to n
      for j: 1 to n
        Find minimum value Min from matrix D(Si,Sj)
          if (Min>=Threshold)
            return
      End for
    End for
      Merge(S_i,S_j)
      Update Dissimilarity matrix D
      Decrement Count by 1
end
end
```

Bottom-up methods are used in hierarchical agglomerative algorithms, which consider each item in its own cluster and then combines two clusters at a time until all clusters are merged into a single cluster. In this paper, each session with set of navigated pages considered as single cluster. The minimal value determined from Dissimilarity matrix D, which is derived using Equation (3), is used to merge two clusters. As a result, the sessions that are extremely similar are combined first. The Dissimilarity matrix is recalculated after each merge process. When the minimum distance between sessions reaches a specific Threshold value, the process ends. To acquire the appropriate number of clusters, Threshold values are modified considerably.

Consider a dissimilarity matrix with 5 sessions to demonstrate how the hierarchical agglomerative algorithm works.

Table 3 Sample dissimilarity matrix

	S_1	S_2	S_3	S_4	S_5
S_1	0	2.8	1.33	2.5	3.6
S_2	2.8	0	3.3	2.6	1.8
S_3	1.33	3.3	0	3	3.5
S_4	2.5	2.6	3	0	3.1
S_5	3.6	1.8	3.5	3.1	0

The algorithm examines the entire matrix for the minimum value, in this example it is between the session S_1 and S_3, which is 1.33. This is the smallest value, and it represents a pair of sessions that are quite close to one other.

Now combine the two sessions and recalculate the dissimilarity as indicated in Table 4.

Table 4 Recalculated dissimilarity matrix

	S_1,S_3	S_2	S_4	S_5
S_1,S_3	0	2.8	2.5	3.5
S_2	2.8	0	2.6	1.8
S_4	2.5	2.6	0	3.1
S_5	3.5	1.8	3.1	0

To find the distance between S_2 with $\{S_1, S_3\}$, compute Min (D (S_2, $\{S_1$, $S_3\}$) by referring the values of previous table.

$$\begin{aligned} \text{Min}(\text{D}(\text{S}_2, \{\text{S}_1, \text{S}_3\}) &= \text{Min}(\text{D}(\text{S}_2, \text{S}_1), \text{D}(\text{S}_2, \text{S}_3)) \\ &= \text{Min}(2.8, 3.3) \\ &= 2.8 \end{aligned}$$

Similarly, the distance between S_4, S_2, S_3 and S_5, S_2, S_3 must be calculated. The updated Dissimilarity matrix can be used to calculate the minimum value in the next round, and the process can continue until the termination condition is met.

Algorithm: Adapted K-means Clustering

Input: user sessions
Output: set of clusters
Randomly select any k sessions as cluster centroids. M=$\{m_1,m_2,\ldots m_k\}$
Assign $\{m_i = S_i\}$, where 0<i<K+1
For each session S_j, 0< j<n, n is the cumulative session count
repeat
Calculate D = $\{d_1,d_2,d_3\ldots d_k\}$, where dj=DESW($S_j$,$m_i$), and $0 < i <$ K+1, $0 < j <$ n+1
Find d_i:=minimum(D), where $0 < i <$ K+1 for each Ci,
Assign session S_j to cluster C_i based on d_i.
Redo for each cluster C_i
Find the session S_p such that DESW(S_p,S_r) is minimum, where $S_p \epsilon C_i$, and $S_r \epsilon C_i$, 0<r<number of session in cluster C_i
Now select new cluster centroids, $\{m_i = S_p\}$, where 0<i<K+1, 0<p< total number session in each cluster
Until no new cluster centroids are found

Function: DESW

Input: Two S_i and S_j Web sessions
Output: Distance from session S_i to S_j
For S_i and S_j find N_p i.e. Total number of pages visited by the user
Find weight of the page visited by the user in session S_i and S_j
$S_i(W(P_k))$ and $S_j(W(P_k))$, where $W(P_k) = \{W(P_1), W(P_2), \ldots W(P_k)\}$, where $P_1, P_2, \ldots P_k$ are web pages
Compute D $(S_i, S_j) = \frac{1}{NP} \sum_{k=1}^{Np} |S_i(W(P_k)) - Sj(W(P_k))|$
Return D

K cluster centroids are randomly selected using an adapted K-means clustering algorithm. Sessions are assigned to the cluster centroid closest to them. After the clusters have been constructed in this manner, the cluster centroid is chosen as the session that is closest to all the other sessions in the cluster. Once the clusters are formed this way, a session which is closet to all the other session in a same cluster is selected as cluster centroid. This process continues until no new cluster centroids are found. Because of the innovative distance function derived using Equation (3), this updated technique produces superior cluster quality.

7 Evaluating Performance of Clustering Algorithm

Cluster evaluation is used to determine the optimal count of session clusters using various algorithms for clustering. The objective of Dunn index and Davies–Bouldin Index is to locate group of clusters that are tightly packed, with minor variance between cluster members, and well partitioned. In this research work, cluster evaluation measure applied on two clustering algorithm namely adapted K-Means, Hierarchical to identify the optimal numbers of clusters.

7.1 Dunn Index

Here, Dunn Index [22] is used as cluster evaluation measure which gives result on the basis of the lowest distance amongst objects in distinct clusters and the greatest distance across objects inside the same cluster. Dunn Index value ranges from 0 to ∞. Let $D_{\min}$ be the lowest distance across objects in different clusters and $D_{\max}$ be the greatest distance in the same cluster. The distance between clusters C_i and C_j is measured by the distance between their

nearest points

$$\mathrm{Dist}(\mathrm{C_i}, \mathrm{C_j}) = \min_{xi \in Ci, xj \in Cj} d(xi, xj) \tag{5}$$

The equation finds minimum distance between clusters $\mathrm{C_i}$ and $\mathrm{C_j}$, where $\mathrm{x_i} \in Ci$ and $\mathrm{x_j} \in Cj$. Dmin finds the smallest distance of these.

$$Dmin = \min_{i \neq j} Dist(Ci, Cj) \tag{6}$$

For each cluster Ck, lets us take diameter as Diam(Ck), which represent largest distance between the two points in the same cluster.

$$\mathrm{Diam}(\mathrm{C_k}) = \max_{xi, xj \in Ck} d(xi, xj) \tag{7}$$

Then Dmax is the larget of these distances Diam($\mathrm{C_k}$).

$$Dmax = \max_{1<k<K} \mathrm{Diam(Ck)} \tag{8}$$

The Dunn Index is described as ratio between $\mathrm{D_{min}}$ and $\mathrm{D_{max}}$.

$$\mathrm{DI} = \frac{Dmin}{Dmax} \tag{9}$$

7.2 Davies–Bouldin Index

Davies–Bouldin Index (DBI) [22] is widely used validity measure to identify compactness and well separation of the cluster. Let K is the total cluster count, $\mathrm{C_i}$ be the cluster centre of i, σi is the estimated distance between each object in the $\mathrm{i^{th}}$ cluster and the $\mathrm{C_i}$ center of the cluster and the gap between the $\mathrm{C_i}$ and $\mathrm{C_j}$ cluster centres is expressed by d($\mathrm{C_i}$,$\mathrm{C_j}$). Now DBI is denoted as

$$\mathrm{DBI} = \frac{1}{K} \sum_{i=1}^{k} \max_{i \neq j} \frac{(\sigma_\mathrm{i} + \sigma_\mathrm{j})}{d(C_i, C_j)} \tag{10}$$

8 Results and Discussion

This segment discusses the output obtained in various phases of the web usage mining system. Web server log file is downloaded from http://www.almhuette-raith.at/apache-log/access.log, which contain 9874 records. It carries web browsing entry from 12th December 2015 to 21st December 2015. Experiments are conducted in Java Eclipse platform using Java programming language.

Data cleaning operation is applied on raw web log file to remove irrelevant or noisy data. This stage removes all URL entries for robots, multimedia files and error status code. Thus, all insignificant URLs are removed from the log file. Result of data cleaning operation reduced the number of entries from 9874 to 5104. Table 5 shows the number of entries in the original web log and cleaned file type and format of web log. Table 6 depicts the number of entries for script, multimedia and image file.

Table 5 Web log file specifications pre and post cleaning

Size of Original Web Log	No. of Original Log File Entries	Size of Cleaned Web Log	No. of Cleaned Web Data Records	Web Log File Format	Type of Web Log File
1870 KB	9874	967 KB	5104	Combined Log Format	Text file

Table 6 Result of data cleaning process

No. of Days	Original Web Log Data	No of Script File	No of Multimedia File	Number of Image File	Cleaned Web Log Data
10 days	9874	330	79	764	5104

Now user identification algorithm is executed to identify unique users based on IP address and agent field. This stage identifies various unique users browsed this web site along with number of URLs accessed by them. The semantic based session identification algorithm applied on pre-processed weblog file which discovers web session, based on important criteria such as time, referrer and semantic check.

Table 7 represents the result of number of users; time based user session; Time-referrer based session and modified semantic user session. Even if the log entry belongs to the same session, the time-based method establishes a new session by comparing the page view time and the session time. As a result, more dense sessions are created. To produce a better experience, the Time-Referrer algorithm evaluates both time and referrer information. Before creating a new session, the proposed approach checks for semantic distance, time, and referrer. It checks for all of the conditions listed in the algorithm, and if none of them are met, it initiates a new session. As a result, this strategy prevents the formation of fraudulent sessions and adds to the formation of high-quality sessions.

Table 7 Result of user and modified semantic session identification

Cleaned Web Log Data	Number of Users	Traditional Time Based Sessionization	Time-Referrer Based Sessions	Modified Semantic Based Session Algorithm
5104	2116	3758	3576	3312

Table 8 Evaluation of modified semantic session identification of algorithm

No of Real Session	Traditional Time Based		Time-Referrer Based		Modified Semantic Based	
32	No of session	%	No of session	%	No of session	%
	24	75	25	78	27	84

Table 8 shows the comparison of modified Semantic based session identification method with traditional and Time – Referrer based session method. To obtain the accuracy of the proposed algorithm, we have considered small test data containing 32 sessions. These true sessions are taken in to consideration manually by counting. Results clearly represents that accuracy of the modified semantic based method is 84%, which is better than traditional Time based and Time-referrer method.

Table 9 Part of dissimilarity matrix assessed on the basis of session weight

D(Si, Sj)	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9
S_1	0	26	26	0	0	14	0	9	9
S_2	26	0	26	0	0	0	0	9	9
S_3	26	26	0	0	0	0	0	9	9
S_4	0	0	0	0	9	0	0	9	9
S_5	0	0	0	9	0	0	0	9	9
S_6	14	0	0	0	0	0	0	9	9
S_7	0	0	0	0	0	0	0	9	9
S_8	9	9	9	9	9	9	9	0	9
S_9	9	9	9	9	9	9	9	9	0

To cluster user sessions, first step in clustering process is computation of page weight and session weight as mentioned in the Equation (2). Then dissimilarity matrix is computed using Equation (3). Table 9 represents the distance between the sessions as per the computations based on page and session weight. The whole dissimilarity matrix is taken as input for both modified K-means and Hierarchical algorithm. Since the sessions are clustered based on the minimum dissimilarity value, the accuracy of the clusters formed using both algorithm are better compared to applying clustering

algorithm directly on the sessions obtained using traditional session algorithm. Experimental results given in Figures 2 and 3 represents accuracy of the algorithms which are measured using Dunn Index and Davies-Bouldin Index. These Indices identify set of clusters that are packed together and well separated. Cluster quality will be better for higher the value of Dunn Index and lower the value of DBI.

Figures 2 and 3 represents accuracy of the adapted Hierarchical agglomerative algorithm using DI and DBI. Figures 4 and 5 compares the accuracy of adapted K-means Clustering algorithm with respect to Semantic based and Time-referrer based web sessions. It demonstrates that accuracy of proposed algorithm is better for semantic based sessions than the existing algorithm.

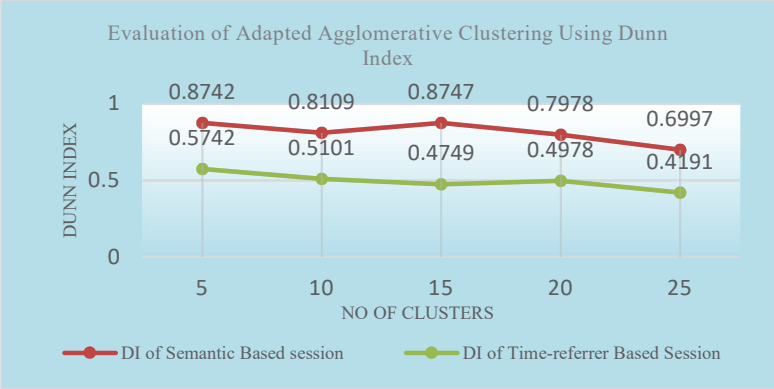

Figure 2 Evaluation of adapted agglomerative clustering using Dunn index.

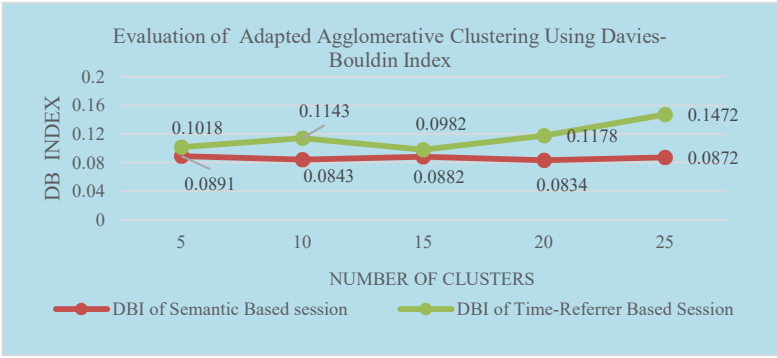

Figure 3 Evaluation of adapted agglomerative clustering using Davies-Bouldin index.

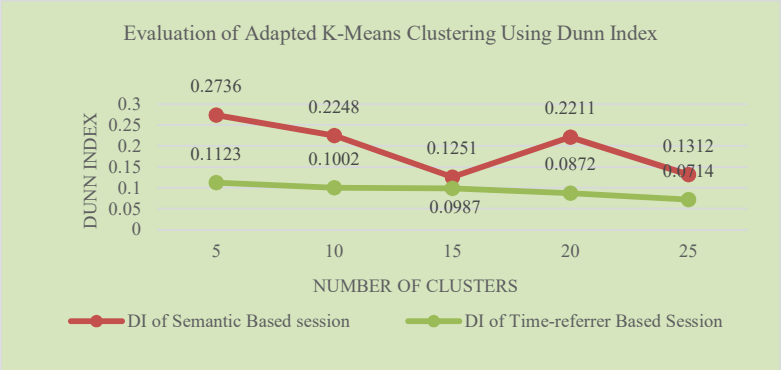

Figure 4 Evaluation of adapted K-Means clustering using Dunn index.

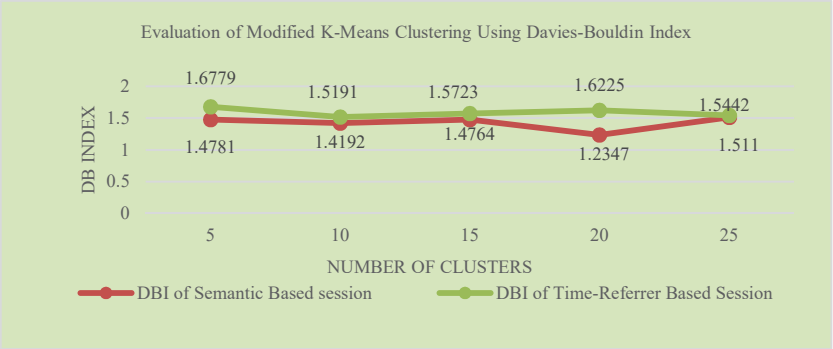

Figure 5 Evaluation of adapted K-means clustering using Davies-Bouldin index.

9 Conclusion

In this article, the suggested semantic distance-based approach solves the main drawbacks of conventional session identification algorithms, such as the production of dense sessions. To generate a new session, all three criteria are checked, including semantic distance between URLs, time, and referrer. The author's suggestion to compute semantic distance contributed to the construction of high-quality sessions. Each user-visited web page is allocated a weight based on the length of time the user spent on the page. It clearly demonstrates the visitor's attention on the website. This is then used to determine session weight, which is subsequently used to assess dissimilarity between sessions using both page weight and session weights as inputs.

The entire set, represented as a dissimilarity matrix, is then fed into both a modified agglomerative and an adapted K-means clustering method, resulting in more precise cluster creation. Experimental results shows that semantic based sessionization along with the assignments of weight to web pages and user sessions have been found to improve the efficiency of the suggested clustering algorithm and its consistency the terms of DI and DBI indexes. The aim of the weight allocation to page and session was to mitigate the negative effect of irrelevant web sessions and URLs leading to the creation of value added clusters. In future we will work on parallel processing technique to improve time efficiency of session and clustering algorithm on web usage data.

References

[1] D.S. Anupama, S.D. Gowda, 'Clustering of web user sessions to maintain occurrence of sequence in navigation pattern', Second International Symposium on Computer Vision and the Internet (VisionNet'15), Elsevier, 2015.

[2] M. Munk, M. Drlík, and J. Reichel, 'Quantitative and Qualitative Evaluation of Sequence Patterns Found by Application of Different Educational Data Preprocessing Techniques', IEEE, 2017.

[3] S. P. Mary, E. Baburaj, 'Performance Enhancement in Session Identification', International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), IEEE, 2014.

[4] M. Srivastava, R. Garg, P.K. Mishra, 'A MapReduce-Based User Identification Algorithm in Web Usage Mining', International Journal of Information Technology and Web Engineering, Volume 13, April–June 2018.

[5] K. Mihara, M. Terabe and K. Hashimoto, 'A Novel Web Usage Mining Method Mining and Clustering of DAG Access Patterns Considering Page Browsing Time', International Conference on Web Information Systems and Technologies, 2008.

[6] R. Katarya, O. P. Verma, 'An effective web page recommender system with fuzzy c-mean clustering', Springer, October 2016.

[7] G. Poornalatha and P. S. Raghavendra, 'Web User Session Clustering Using Modified K-Means Algorithm', International Conference on Advances in Computing and Communications, Springer, Berlin, Heidelberg, 2011.

[8] D. Xu, Y. Tian, 'A Comprehensive Survey of Clustering Algorithms', Springer Berlin Heidelberg, Annals of Data Science, 2015

[9] V. Bhatnagar, R. Majhi, P. R. Jena, 'Comparative Performance Evaluation of Clustering Algorithms for Grouping Manufacturing Firms', Arab J Sci Eng.

[10] N. Kaur, Dr. H. Aggarwal, 'A Novel Semantically-Time-Referrer based Approach of Web Usage Mining for Improved Sessionization in Pre-Processing of Web Log', (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 8, No. 1, 2017.

[11] Z. A. Ansari, A.S. Syed, 'Discovery of Web Usage Patterns Using Fuzzy Mountain Clustering", Int. J. Business Intelligence and Data Mining, Vol. 11, No. 1, 2016.

[12] P. Nerurkar, A. Shrike, M. Chandane, S. Bhirud, 'Empirical Analysis of Data Clustering Algorithms', 6th International Conference on Smart Computing and Communications, ICSCC 2017.

[13] P. Dhanalakshmi, K. Ramani, E. Reddy, 'The Research of Preprocessing and Pattern Discovery Techniques on Web Log File', IEEE 6th International Conference on Advanced Computing (IACC), 139–145, 2016.

[14] G. D. Kumar and M. Gosul, 'Web Mining Research and Future Directions', Advances in Network Security and Applications, Volume 196, 2011.

[15] S. Miyamoto, 'An Overview of Hierarchical and Non-hierarchical Algorithms of Clustering for Semi-supervised Classification', International Conference on Modeling Decisions for Artificial Intelligence, MDAI 2012, vol. 7647. Springer, Berlin, Heidelberg.

[16] O. Sangita, J. Dhanamma, 'An Improved K-Means Clustering Approach for Teaching Evaluation', Advances in Computing, Communication and Control, vol. 125, Springer, Berlin, Heidelberg, 2011.

[17] M. Srivatsava, R. Garg, 'Analysis of Data Extraction and Data Cleaning in Web Usage Mining', Analysis of Data Extraction Data Cleaning in Web Usage Mining, Proceedings of th International Conference on Advanced Research in Computer Science and Engineering, 2015.

[18] K. S. Reddy, G. P. Saradhi Varma, 'Preprocessing the web server logs: an illustrative approach for effective usage mining', ACM SIGSOFT Software Engineering Notes, May 2012, Volume 37 Number 3.

[19] P. Sengottuvelan, Lokeshkumar, 'Session Identification in Web Usage Mining to Personalize the Web', International Journal of Applied Engineering Research, ISSN 0973-4562, Volume 10, Number 9, 2015.

[20] J. Kapusta, 'User Identification in the Process of Web Usage Data Preprocessing', International Journal of Emerging Technologies in Learning (iJET), Vol. 14, No. 9, 2019.

[21] N. Goel, C.K. Jha, 'Preprocessing Web logs: A Critical phase in Web Usage Mining', International Conference on Advances in Computer Engineering and Applications (ICACEA), 2015.
[22] M.Z. Rodriguez, C.H. Comin, D. Casanova, O.M. Bruno, D.R. Amancio, Costa LdF, 'Clustering algorithms: A comparative approach', 2019.
[23] W. Lu, 'Improved K-Means Clustering Algorithm for Big Data Mining under Hadoop Parallel Framework', J Grid Computing 18, 239–250, 2020.
[24] A. Sinha, 'A hybrid MapReduce-based k-means clustering using genetic algorithm for distributed datasets', J Supercomput 74, 1562–1579, 2018.
[25] O. Kassak, M. Kompan, M. Bielikova, 'Acquisition and Modelling of Short-Term User Behaviour on the Web: A Survey', Journal of Web Engineering, Vol. 17(5), 23–70, 2018.
[26] http://www.almhuette-raith.at/apache-log/access.log

Biographies

Sowmya HK received the Bachelor of Engineering degree in Computer Science and Engineering from Kurunji Venkataramana Gowda College of Engineering in 2004, the Master of Engineering degree in Computer Science and Engineering from University Visvesvaraya College of Engineering, Bangalore University in 2010 respectively. She is currently working as Senior Assistant Professor at the Department of Artificial Intelligence and Machine Learning, New Horizon College of Engineering, Visvesvaraya Technological University. She is currently pursuing Ph.D at the Department of Information Science and Engineering, New Horizon College of Engineering, Visvesvaraya Technological University. Her research areas include Web Usage Mining, Data Mining, and Deep Learning.

R. J. Anandhi received the Bachelor of Engineering degree in Computer Science and Engineering from Government College of Technology, Coimbatore in 1991. She secured Master of Technology degree in Computer Science and Engineering from Pondicherry Central University in 1995. She pursued Ph.D from Dr. MGR University, Chennai in 2011. She is currently working as Professor and Head of the Department of Information Science & Engineering at New Horizon College of Engineering, Bengaluru. Her research areas include Data Mining, NLP and Cloud Computing.