International Journal of Engineering & Technology, 7 (4) (2018) 3154-3161

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.19145
Research paper



Clustering web users for reductions the internet traffic load and users access cost based on K-means algorithm

Maged Nasser [1] *, Naomie Salim [1], Hentabli Hamza [1], Faisal Saeed [2]

[1]School of Computing, University Technology Malaysia, Johor Bahru, Johor 81310, Malaysia
[2] College of Computer Science and Engineering, Taibah University, Medina, Saudi Arabia
**Corresponding author E-mail: maged.m.nasser@gmail.com*

Abstract

The continuous growth in the size and use of the Internet is increasing the difficulties in searching for information. Reductions on the Inter-net traffic load and user access cost is therefore particular important. Clustering is an important part of web mining that involves finding natural groupings of web resources or web users. Researchers have pointed out some important differences between clustering in conven-tional applications and clustering in web mining. Web clustering as an important web usage mining (WUM) task groups web users based on their browsing patterns to ensure the provision of a useful knowledge of personalized web services. Based on the web structure, each Uniform Resource Locator (URL) in the web log data is parsed into tokens which are uniquely identified for URLs classification. The col-lective sequence of URLs a user navigated over a period of 30 minutes is considered as a session and the session is a representation of the users' navigation pattern. This paper proposes a variation of the K-means clustering algorithm based on properties of rough sets. The pro-posed algorithm represents the clustering of the web users based on their browsing activities or patterns on the web. Specifically, a user may visit a website often and spends much time on each visit. users with similar browsing activities are clustered or grouped in to clusters. The paper also describes the design of an experiment including data collection and the clustering process.

Keywords: *Web User Clustering; K-Means; Vector Matrix; Similarity.*

1. Introduction

Clustering is the process of grouping data into disjoint set called clusters such as that similarities among data members within the same cluster are maximal while similarities among data members from different clusters are minimal. The optimization of this criterion is nondeterministic polynomial time (NP) hard problem in general Euclidean space d, even when the clustering process deals with only two clusters[1]. Clustering the document in data mining is one of the traditional approaches in which the same documents that are more relevant are grouped together. Document clustering take part in achieving accuracy that retrieve information for systems that identifies the nearest neighbors of the document. Day to day the massive quantity of data is being generated and it is clustered [2]. K-means algorithm is the easiest way of learning algorithm to handle and to solve the generally known problem of grouping. Aims at partition to a group of objects based on their attributes into k groups which is user predefined constant [2]. The main goals of a web data clustering algorithm are to produce appropriate clusters for the end user, to assign the available data to the most relevant cluster, to respond the end user instantly [3].

The Internet has become a major means of life, work, study, and information dissemination. Numerous organizations are providing web-based services due to the consistent increase in web development and the number of available web searching tools. However, information management is becoming troublesome due to the continuous growth in the use and size of the Internet. Hence, there is a need to develop new techniques to improve web performance[4].

Web mining refers to the intelligent analysis of web data; it helps organizations to have a better knowledge of the choices of the web users and help them to run their requirements more efficiently [5]. The clustering of web users based on their similarities is one of the web mining techniques. The web designer can get a better knowledge of the user preferences by analyzing the characteristics of each cluster. This analysis will help in the provision of more suitable and customized services [6].

Virtually all the web clustering methods are based on the similarity of users interests and access patterns; they cluster based on the outcome of these measures. Mining the history of a users' access patterns do not only provide information on the web usage, it also provides some behavioral traits of web users [7]. The need to understand web-users has gained more interest in recent times due to the recent web advancements and the proliferation in the number of web-based applications. Web users can be clustered based on different criteria and useful knowledge can be derived from their access pattern [8]. The knowledge gained can also help in the management of many applications [9]. One of such applications is the prefetching of web pages to assist in personalizing the needs of the user and minimize their waiting time [10]. The other applications may include proxy cache organization [11], [12] and mapping of user access patterns [3]. Few web clustering methods exist [8], [13], [14]; however, their direct application on the primitive user access data is not efficient and fail to establish exciting clusters since web server may often contain several pages which web users may access with different interests [8]. This clustering in this study is focused on the users' navigation pattern. Specifically, a user may visit a website often and spends much time on each visit. The concept of session was introduced to deal with the unpredictable nature of web browsing; it was introduced to serve as the unit

Copyright © 2018 Maged Nasser et. al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

of interaction between a web server and a user [14]. The clustering of the user's browsing sessions can help a web developer to understand the browsing pattern of the users and help in the provision of more user-specified services. This knowledge can also contribute to the construction and maintenance of intelligent real-time web servers with dynamic designs to suit future users' needs [15].

Clustering algorithms are used to divide objects into clusters and subsets, with the aim of creating clusters that are internally coherent, but clearly distinct from each other [16]. This implies that objects within a cluster must be as similar as possible and should differ considerably from those in the other clusters. There are several available clustering methods and each of them groups datasets differently. The clustering methods are selected based on the type of output intended, as well as on the known performance of method with the available type of data [16].

The hard clustering algorithms are exhaustive (can assign each object to some cluster) or non-exhaustive (some objects may not be clustered). The hard clustering algorithms can either be flat and hierarchical; the goal of the flat clustering algorithms is to divide the object space into several clusters in that each cluster consists of similar objects and different from the content of the other clusters[17]. Then, the K-mean algorithm is used to compute the similarity between the objects before clustering them. The K-means algorithm was used in this study to compute the similarity between all the web users before clustering them based on their similarity [17].

2. Related works

The field of web usage mining (WUM) has recently become an active commercialization and research area. The WUM mainly aims to average the data sourced from users' interactions with the web in order to model patterns that are important for web personalization [18]. Some of the current methods for web usage data mining are statistical analysis, sequential patterns, association rules, classification, and clustering [19-23]. An important WUM topic is web users clustering which involves the discovery of the user clusters with similar information needs, such as users that visits similar web pages. The analysis of the clusters' characteristics can help web developers to understand the user's patterns and be in a better position to provide more customized and suitable services.

A comprehensive method in which users' sessions are clustered, evaluated, and interpreted has been presented by Pallis et al. [24]. Xiao et al. also proposed a method of measuring the similarity among the interests of web users based on their past access patterns [25]. linHuaXu et al. studied and clustered the behavior of web users [26] and the findings of the study showed the efficiency and feasibility of using such algorithms. The clustering was done based on a vector matrix and k means algorithm. Also, k-mean used in many different area for many issue. Techniques for the improvement of the k-means algorithm by finding fixed centroids and applying a clustering framework to produce similar clusters for each run have been proposed by Chaitraa et al.[27]. Furthermore, Poornalatha et al. suggested the improvements of the K-means algorithm and its application in web sessions clustering [28]. This method addressed the differences in the length of sessions. Duraiswamy et al. proposed the use of matrix for the calculation of sessions' similarity before using agglomerative hierarchical clustering algorithm for the clustering [29].

Previously, we defined the levels of web users' similarities to establish the interests of different web users. However, this definition depends on the application and its function could be based on the number of visits to a page or the number of times a page was visite[30]; it may also depend on visiting the orders of links. Later, two users that visited a page could be clustered into different groups with different interests if they visit the pages in a particular order. A matrix-based framework has been developed for clustering web users in a way that closely related users are clustered together based on their similarity measure [3]. However, an increase in the number of users deteriorated the performance of the clustering method, especially when a threshold number has been reached. Moreover, a page may be visited by the same user several times using either the same or different routes [31]; this makes it difficult to establish the visiting pattern based on similarity.

3. Motivation and problem statement

The rapid web development and the increased number of available web searching tools push more and more organizations into putting their information on the web to provide web-based services. In the meantime, the continuous growth in the size and use of the Internet is increasing the difficulties in searching for information. Reductions on the Internet traffic load and user access cost are therefore particularly important. An important attribute contributing to the popularity of a website is the degree of personalization it offers when presenting its services to users. However, improving the level of user personalization by reorganizing the entire website structure according to the interests of each user increases the number of computations at the web server hosting the website. One solution to avoid this problem is to group users based on their web interests, and then, organize the structure of the website in a manner suitable to the web needs of different groups.

4. Methodology and experiment results

An important step prior to experimentation in web mining is data preprocessing. The aim of this step is to transform the log data into a suitable format to be analyzed depending on the needs of the analysis. Many steps have been done for this work during data preprocessing and data clustering as it shown in the Methodology process in Fig. 1, the aim is to meet the demands of the current mining task and as such, the data preprocessing step in our analysis consists of four steps which are data cleaning, users' identification, session identification, and vector matrix. The data cleaning step involves the removal of the redundant data from the dataset while the user identification step involves knowing the users that visited a website. The session identification step aims to divide the page accesses of each user into individual sessions and this can be easily achieved through a timeout, while vector matrix implies getting the hits information of each users' access to different pages.

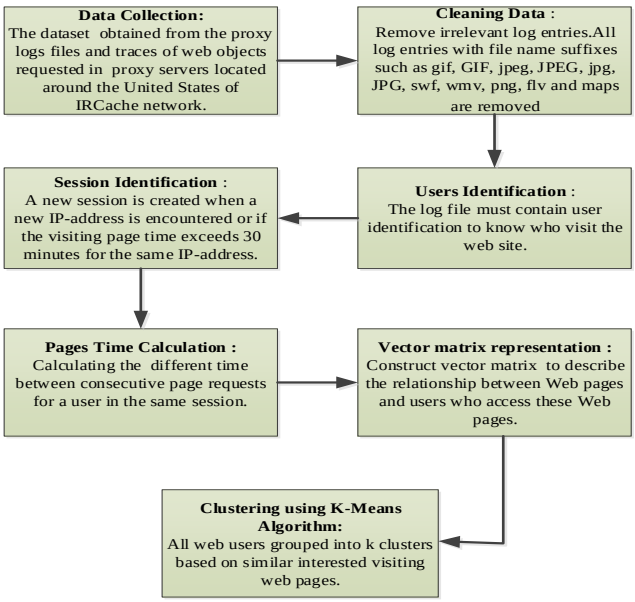


Fig. 1: Methodology of Process.

4.1. Dataset

The web proxy logs file records information of users' accesses to many web objects in the past. The proxy logs file reflects the behaviors of users and can be considered as a complete and prior knowledge of users' interests. Thus, web proxy logs file can be exploited for predicting future requests. Fig. 2 shows a sample of proxy logs file, which is used for clustering the users based on visited pages.

1282348821.049 138 132.55.200.134 TCP_MISS/301 471 GET http://dishnetwork.com/ - DIRECT/205.172.147.51 text/html
1282348917.977 286 249.78.126.183 TCP_MISS/302 1753 GET http://www.hotmail.com/ - DIRECT/64.4.20.184 text/html
1282348954.666 393 132.55.200.134 TCP_MISS/200 22321 GET http://www.msn.com/ - DIRECT/65.55.17.26 text/html
1282348982.398 200 249.78.126.183 TCP_MISS/200 6550 GET http://www.btt.com.ar/foto/t/12/75/1275530489_mark-webb2.jpg - DIRECT/72.232.178.138 image/jpeg
1282348982.437 286 249.78.126.183 TCP_MISS/200 22520 GET http://www.btt.com.ar/foto/t/12/81/1281356302_DSC03379.JPG - DIRECT/72.232.178.138 image/jpeg
1282349600.500 78 50.83.47.141 TCP_MISS/200 344 POST http://app.ninjasaga.com/amf/ - DIRECT/75.126.166.176 application/x-amf
1282361594.519 604 171.11.238.157 TCP_MISS/200 2209 GET http://nt0.ggpht.com/news/tbn/dNDwpcYNGpFYaM/0.jpg - DIRECT/74.125.153.103 image/jpeg
1282361594.561 288 171.11.238.157 TCP_MISS/200 1772 GET http://nt3.ggpht.com/news/tbn/N-iKYy_kXVXPuM/0.jpg - DIRECT/74.125.153.104 image/jpeg
1282411485.94410 60.247.185.54 TCP_MEM_HIT/200 834 GET http://videos.asianbabemedia.com/jp18babydoll.asx - NONE/- video/x-ms-asf

Fig. 2: Shows the Proxy Logs File.

An access proxy log entry of the proxy logs file is usually composed of 10 fields, including timestamp, client address, elapsed time, request method, URL, size in bytes, user identification, hostname, log tag and HTTP code, hierarchy data, and content type. The data used in this study was sourced from the proxy logs files and web object traces which are requested in BO2 proxy servers hosted around the United States of IRCache network for one day [32] . The meanings of the ten fields for each log entry of the proxy logs file are given in Table 1.

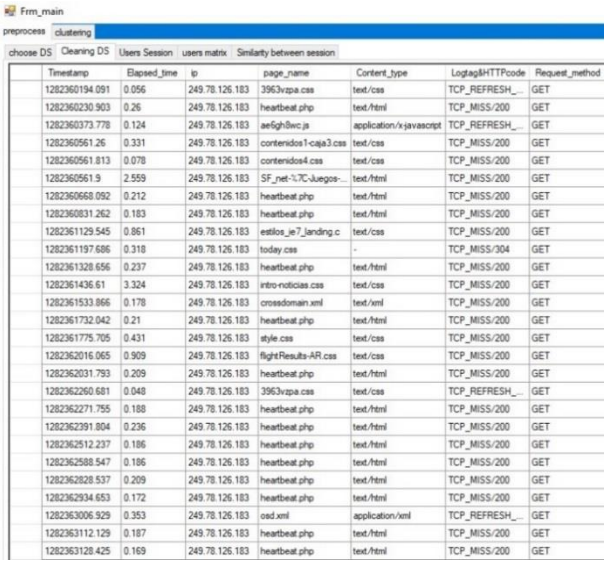
Table 1: Exeplanation of the Fields of Log Entry in the Proxy Logs File

Field	Meaning
Timestamp	The time when the client socket is closed. The format is "Unix time" (seconds since Jan 1, 1970) with millisecond resolution.
Elapsed time	The elapsed time of the request, in milliseconds.
Client address	A random IP address identifying the client.
Log tag and HTTP code	The log tag describes how the request was treated locally (hit, miss, etc).But the HTTP status code is the reply code taken from the first line of the HTTP reply header.
Size	The number of bytes written to the client
Request method	The HTTP request method.
URL	The requested URL.
User identification	Always '-' for the IRCache logs.
Hierarchy data and hostname	A description of how and where the requested and Hostname object were fetched.
Content type	The content-type field from the HTTP reply.

4.2. Data cleaning

This step involves the application of several filtering techniques to ensure the removal of redundant log entries. Being that both scripts and graphics are downloaded together with Hypertext Markup Language (HTML) file, a user's request to access a page may often result in the generation of several log entries [13]. As the aim of WUM is to extract the users' pattern based on previous behaviors, there is no need to include file requests that are not explicitly requested by the user [33]. All log entries with file name suffixes such as gif, GIF, jpeg, JPEG, jpg, JPG, swf, wmv, png, flv, and maps are removed during data cleaning. Data cleaning demands a trace preparation step during which the irrelevant or redundant requests (such as uncatchable and dynamic requests) are cleaned from the log files.

Unnecessary fields should be removed like size, hierarchy data and hostname and user identification because the value of this field always is '-'that does not give any information. The result of reading data set observed the number of log files was 37661 and after cleaned the dataset the number of log files decreased into 3446. Fig. 3 show the pats of these files.



	Timestamp	Elapsed_time	ip	page_name	Content_type	Logtag&HTTPcode	Request_method
	1282360194.091	0.056	249.78.126.183	3963vzpa.css	text/css	TCP_REFRESH_...	GET
	1282360230.903	0.26	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282360373.778	0.124	249.78.126.183	ae6gh8wc.js	application/x-javascript	TCP_REFRESH_...	GET
	1282360561.26	0.331	249.78.126.183	contenidos1-caja3.css	text/css	TCP_MISS/200	GET
	1282360561.813	0.078	249.78.126.183	contenidos4.css	text/css	TCP_MISS/200	GET
	1282360561.9	2.559	249.78.126.183	SF_net-%7C-Juegos-...	text/html	TCP_MISS/200	GET
	1282360668.092	0.212	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282360831.262	0.183	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282361129.545	0.861	249.78.126.183	estilos_ie7_landing.c	text/css	TCP_MISS/200	GET
	1282361197.686	0.318	249.78.126.183	today.css	-	TCP_MISS/304	GET
	1282361328.656	0.237	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282361436.61	3.324	249.78.126.183	intro-noticias.css	text/css	TCP_MISS/200	GET
	1282361533.866	0.178	249.78.126.183	crossdomain.xml	text/xml	TCP_MISS/200	GET
	1282361732.042	0.21	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282361775.705	0.431	249.78.126.183	style.css	text/css	TCP_MISS/200	GET
	1282362016.065	0.909	249.78.126.183	flightResults-AR.css	text/css	TCP_MISS/200	GET
	1282362031.793	0.209	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362260.681	0.048	249.78.126.183	3963vzpa.css	text/css	TCP_REFRESH_...	GET
	1282362271.755	0.188	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362391.804	0.236	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362512.237	0.186	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362588.547	0.186	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362828.537	0.209	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282362934.653	0.172	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282363006.929	0.353	249.78.126.183	osd.xml	application/xml	TCP_REFRESH_...	GET
	1282363112.129	0.187	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET
	1282363128.425	0.169	249.78.126.183	heartbeat.php	text/html	TCP_MISS/200	GET

Fig. 3: Dataset after Cleaning.

4.3. User identification

To reveal the identity of a person that visited a web page, the ID of the person such as the login details must be contained in the log file. However, users are not required to log into some websites, while most web servers do not request for user's login ID when using personal computers. Thus, there is not enough information as per the HTTP standard to distinguish among web users from the same proxy or host. Often, such information is the IP address provided by the Internet Service Provider (ISP) or a corporate proxy server to a user's TCP/IP connection to the site, hence, preventing unique identification [34].

4.4. Session identification

Several recommendation systems use the user session in the log file to identify the interests of the users, but they ignore the sequential information about each user session [6]. A session refers to the time an activity was initiated to the time it ended. As per W3C, a session refers to the sum of all the activities performed by a user from the time of logging into a site to the time of exit [35]. Due to the fact, there is no official login and logout to access and use most of the Web sites. It is not very clear when a session begins and ends. Since page request from other servers are not typically available and a user may visit a site more than once.

Session identification is aimed at dividing each users' page accesses into individual sessions and this can be easily achieved through a timeout. With a timeout, a user is assumed to have started a new session if the time between page requests exceeds a certain limit. A new session is automatically launched if new IP-address is found or if the visiting page time allowed for a particular IP-address has been exceeded.

In many commercial products this timeout has been rounded up to 30 minutes. In this study we set 30 minutes. A new session is created when a new IP-address is encountered or if the visiting page time exceeds 30 minutes for the same IP-address. address as

shown in Fig.4. Web pages for each user must be reorder based on time visiting before session identification process.


Frm_main

preprocess | clustering

choose DS | Cleaning DS | Users Session | users matrix | Similarity between session

	No_session	ip	timestamp	Name_page	Elapsed_time
	3	50.83.47.141	1282350982.984	connect.php	0.264
	3	50.83.47.141	1282350989.807	pixel;r=1440723746;fpan=0;fpa=P0-44823...	0.127
	3	50.83.47.141	1282350991.862	pixel;r=1123117432;fpan=0;fpa=P0-79448...	0.126
	4	50.83.47.141	1282353710.609	5eo1yqin.css	0.039
	4	50.83.47.141	1282353710.715	eiim0l2e.css	0.008
	4	50.83.47.141	1282353710.833	7mylnyt.css	0.007
	4	50.83.47.141	1282353710.948	bw7hgwyd.css	0.007
	4	50.83.47.141	1282353711.067	8p5o0tpx.css	0.006
	4	50.83.47.141	1282353711.205	7otigqbu.css	0.008
	4	50.83.47.141	1282353711.323	6b4rklkz.css	0.008
	4	50.83.47.141	1282353711.443	3963vzpa.css	0.008
	4	50.83.47.141	1282353711.573	4wj244ne.js	0.007
	4	50.83.47.141	1282353715.38	ecsf1ki7.css	0.008
	4	50.83.47.141	1282353715.479	58jq9eu7.css	0.007
	4	50.83.47.141	1282353715.654	version.js	0.056
	4	50.83.47.141	1282353715.752	facemoodsInitiator.js	0.029
	4	50.83.47.141	1282353716.568	jquery-1.3.2_nd.js	0.035
	4	50.83.47.141	1282353725.455	ui.core.js	0.037
	4	50.83.47.141	1282353725.967	ui.tabs_open.js	0.032
	4	50.83.47.141	1282353732.852	ageConfirm.aspx	0.255
	4	50.83.47.141	1282353733.564	FeatureLoader.js.php	0.226
	4	50.83.47.141	1282353745.955	pixel;r=2463933396;fpan=0;fpa=P0-448237...	0.225
	4	50.83.47.141	1282353748.14	graphics.js	0
	4	50.83.47.141	1282353748.338	abg.js	0.021
	5	249.78.126.183	1282352282	IM2lawt.avi	10799.729
	5	249.78.126.183	1282352369.645	heartbeat.php	0.228
	5	249.78.126.183	1282352387.934	heartbeat.php	0.171

Fig. 4: Session Identification of Web Pages.

4.5. Vector matrix for users

Prior to web user clustering based on web logs, a vector matrix for the URL and the user was first constructed and the relationship between the web pages and users that visits these pages was described using a URL-User associated matrix R [26]. Let n and m represent the number of web pages and the number of users respectively, then, the matrix can be represented as:

$$R_{mxn} = \begin{bmatrix} hits(1,1) & hits(1,2) & \cdots & hits(1,j) & \cdots & hits(1,n) \\ hits(2,1) & hits(2,2) & \cdots & hits(2,j) & \cdots & hits(2,n) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ hits(i,1) & hits(i,2) & \cdots & hits(i,j) & \cdots & hits(i,n) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ hits(m,1) & hits(m,2) & \cdots & hits(m,j) & \cdots & hits(m,n) \end{bmatrix}$$

Where 'hits' represents a type of user browsing information. The hits of all the users that accessed the web pages over a given time can be directly extracted. From the matrix, users are viewed as the rows, web pages as columns, and the hits count as the values of the elements of this matrix, that is $hits(i,j)$ as the time spent by user i to access the web page j. The i^{th} row vector $R[i,]$ records the counts of the i^{th} user access to all the web pages over a specified period, while the j^{th}column vector $R[,j]$ records the counts of all users who, over the same period, accessed the j^{th} web page. The Fig.5 Show the result of the relationship between the users and the web pages and how much time the user still in the web page.

In Fig. 5 show part of this vector matrix, 3446 web pages was used; number of users = 27, and in the 3rd row and 3rd column indicates that number of users that visited the page 45 over a toke time of 0.007 sec. After getting the hits information vector matrix, we calculated the similarity between users based of visited pages after that apply the K-means algorithm to cluster the web user into some clusters with different k values.

P43	P44	P45	P46	P47	P48
0.006	0.011	0.007	0.006	0.008	0.227
0.006	0.008	0.007	0.008	0.008	0.226
0	0	0	0	0.048	0
0	0	0.023	0	0	0
0	0	0	0	0	0
0.034	0	0.02	0.044	0	0.21
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0.025	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0.061	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Fig. 5: Part of Vector Matrix.

4.6. Users similarity

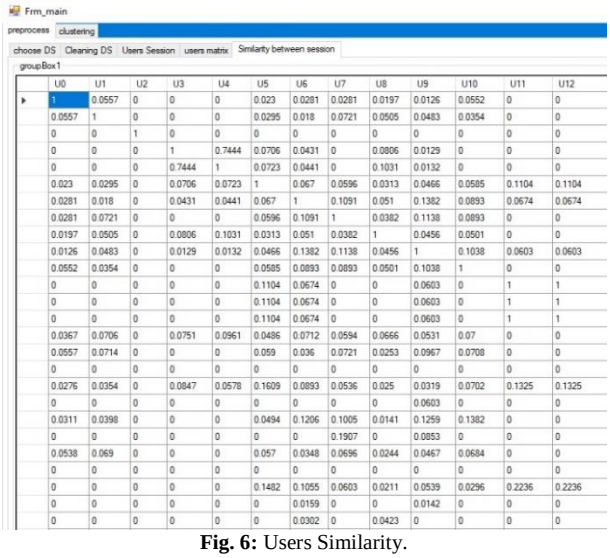
A measure similarity is essential in clustering. based on similarity, clustering is based upon grouping samples. The measure should reflect how close or similar two objects. Suppose that, for a given web site, there are m sessions $S = \{s_1, s_2, \dots, s_n\}$ accessing n different web pages $P = \{p_1, p_2, \dots, p_n\}$ in some time interval. For each page p, and each session s, we associate a usage value, denoted as $use(p,s)$ and defined as:

$$use(P_i, S_j) = \begin{cases} 1 \; if \; P_i is \; accessed \; by \; S_j \\ 0 \; otherwise \end{cases}$$

The $use(*,*)$ vector can be obtained by retrieving the access logs of the site. If two users accessed the same pages in sessions, they might have some similar interests in the sense. They are interested in the same information (e.g., news, electrical products etc). The similarity can be measured by the number of common pages that are accessed. A precise measure of users similarity is to consider the actual period each user spent on each page visited. Let $t(P_k, s_j)$ represent the time the user of session s_j spent on page P_k(assume that $t(P_k, s_j) = 0$) if $s_j$does not include page P_k. Here, the similarity between users can be represented by using cosine similarity [26] as it shown in following formula :

$$Cosine_Sim(s_i, s_j) = \frac{\sum_k (t(P_k, s_i) * t(P_k, s_j))}{\sqrt{\sum_k (t(P_k, s_i))^2 * \sum_k (t(P_k, s_i))^2}}$$

Where $\sum_k (t(P_k, s_i))^2$ represents the square sum of the time the user of session s_i spent while accessing pages at the site, and $\sum_k (t(P_k, s_i) * t(P_k, s_j))$ represents the inner-product over time spent by users of s_i and s_j on visiting the common pages. Even if the same exact pages are visited by two users, their similarity value may be <1 since they spent different times on the page as shown in the fig.6.



	U0	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12
▶	1	0.0557	0	0	0	0.023	0.0281	0.0281	0.0197	0.0126	0.0552	0	0
	0.0557	1	0	0	0	0.0295	0.018	0.0721	0.0505	0.0483	0.0354	0	0
	0	0	1	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	0.7444	0.0706	0.0431	0	0.0806	0.0129	0	0	0
	0	0	0	0.7444	1	0.0723	0.0441	0	0.1031	0.0132	0	0	0
	0.023	0.0295	0	0.0706	0.0723	1	0.067	0.0596	0.0313	0.0466	0.0585	0.1104	0.1104
	0.0281	0.018	0	0.0431	0.0441	0.067	1	0.1091	0.051	0.1382	0.0893	0.0674	0.0674
	0.0281	0.0721	0	0	0	0.0596	0.1091	1	0.0382	0.1138	0.0893	0	0
	0.0197	0.0505	0	0.0806	0.1031	0.0313	0.051	0.0382	1	0.0456	0.0501	0	0
	0.0126	0.0483	0	0.0129	0.0132	0.0466	0.1382	0.1138	0.0456	1	0.1038	0.0603	0.0603
	0.0552	0.0354	0	0	0	0.0585	0.0893	0.0893	0.0501	0.1038	1	0	0
	0	0	0	0	0	0.1104	0.0674	0	0	0.0603	0	1	1
	0	0	0	0	0	0.1104	0.0674	0	0	0.0603	0	1	1
	0	0	0	0	0	0.1104	0.0674	0	0	0.0603	0	1	1
	0.0367	0.0706	0	0.0751	0.0961	0.0486	0.0712	0.0594	0.0666	0.0531	0.07	0	0
	0.0557	0.0714	0	0	0	0.059	0.036	0.0721	0.0253	0.0967	0.0708	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0
	0.0276	0.0354	0	0.0847	0.0578	0.1609	0.0893	0.0536	0.025	0.0319	0.0702	0.1325	0.1325
	0	0	0	0	0	0	0	0	0	0.0603	0	0	0
	0.0311	0.0398	0	0	0	0.0494	0.1206	0.1005	0.0141	0.1259	0.1382	0	0
	0	0	0	0	0	0	0	0.1907	0	0.0853	0	0	0
	0.0538	0.069	0	0	0	0.057	0.0348	0.0696	0.0244	0.0467	0.0684	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0.1482	0.1055	0.0603	0.0211	0.0539	0.0296	0.2236	0.2236
	0	0	0	0	0	0	0.0159	0	0	0.0142	0	0	0
	0	0	0	0	0	0	0.0302	0	0.0423	0	0	0	0

Fig. 6: Users Similarity.

4.7. K-means algorithm

Recently, The K-means is a popular cluster analysis algorithm [36] which was first introduced in 1967 [37]. With this algorithm, the input dataset is partitioned into k different clusters and each sample is assigned to the cluster that has the closest mean. Each cluster is represented by its sample mean and this mean does not necessarily have to be a sample in the dataset. Similarity measure is used based on cosine similarity when comparing users to the clusters. The K-means is an unsupervised learning framework which can solve most of the established clustering problems[38]. It classifies a given data following a simple and easy procedure to a certain number of clusters (assume k clusters) that has already been fixed. The major idea is to define the k centroids, one per cluster, which should be positioned in a cunning way as different locations can give different outcomes. So, it is better to position them far from each other [38].

The next step involves taking each point that belongs to a given data set and associating it to the nearest centroid. Having associated all the points, the first step is done, and an early group age is equally done. Then, the next thing is to recalculate the k new centroids as the barycenters of the clusters from the previous step. Having determined the k new centroids, a new binding will be done between the same data set points and the nearest new centroid, resulting in the generation of a loop [39]. Because of this loop that has been generated, the k centroids may change their location stepwise until there are no more changes (i.e. the centroids are no longer moving). Finally, the k-means algorithm strives to maximize the similarity between points.

The algorithmic flow is as follows:

- Place K points into the space represented by the objects to be clustered (these points are the initial group centroids).
- Group the objects based on the closest centroids.
- Having assigned all the objects, recalculate K centroids' positions with based on the objects in the same cluster.
- Repeat Steps 2 and 3 until the centroids are no longer moving.

Pseudocode fork-means is found in following algorithm. The efficiency of the expectation step of K-means is O (N KD) where N is the number of users in a D-dimension with K clusters.

K-MEANS$\left(\{\vec{X}_1, \ldots., \vec{X}_N\}, K\right)$

1) $\left(\vec{S}_1, \vec{S}_2, \ldots, \vec{S}_K\right) \leftarrow SELECTRANDOMSEEDS\left(\{\vec{X}_1, \ldots., \vec{X}_N\}, K\right)$
2) for K← 1 to K
3) do$\vec{u}_k \leftarrow \vec{S}_K$
4) $w_k \leftarrow \{\}$
5) while stopping criterion has not been met
6) do for n← 1 to N
7) for K← 1 to K
8) j← argMaxSIM($\vec{u}_k$,$\vec{X}_n$)
9) $w_j \leftarrow w_j \cup \{\vec{X}_n\}$
10) for K← 1 to K
11) $\vec{u}_k \leftarrow \frac{1}{|w_k|}\sum_{\vec{x}\in w_k} \vec{X}$ (recomputation of centroids)
12) Return { $w_1, w_2, \ldots, w_{k,}$ }

Where $\{\vec{X}_1, \ldots., \vec{X}_N\}$ *is users vector* , $\left(\vec{S}_1, \vec{S}_2, \ldots, \vec{S}_K\right)$ is the seeds values (initial center value for each cluster), N is the number of users, K is the number of the group clusters and argMaxSIM ($\vec{u}_k$,$\vec{X}_n$) is the max similarity value nearest to the center cluster value.

This Algorithm has been run by using Desktop - C4QOJ2V, Core (TM) i7 CPU, 3. 60 GHZ with 16384MB RAM and Visual c# 2015 has been used for writhing and run this code.

The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters). The main idea is to define k centroids for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The algorithm is implemented when k=2, 3, 4, 5, 6, 7, 8.

In case of using k=2, the result is shown in Fig.7. It is observed that all of user divided into two clusters.

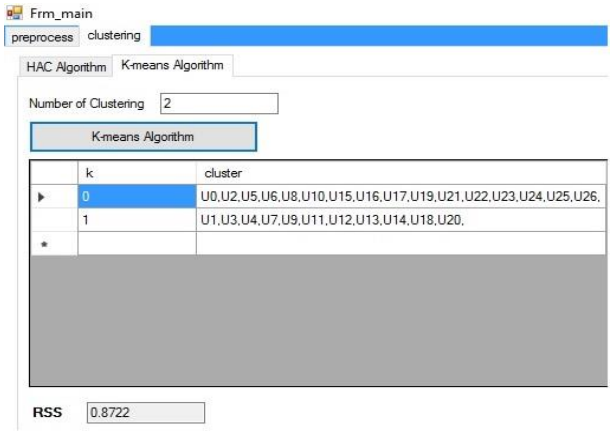


Fig. 7: K-Mean Clustering when K=2.

In case of using k=3, the result is shown in Fig.8. It is observed that all of user divided into three clusters.

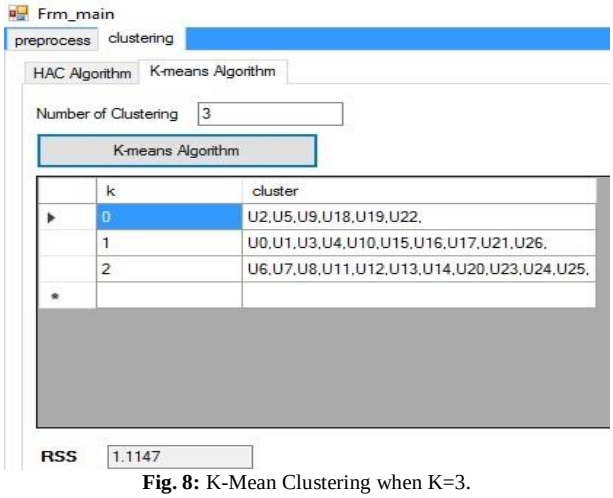


Fig. 8: K-Mean Clustering when K=3.

In case of using k=4, the result is shown in Fig.9. It is observed that all of user divided into four clusters.

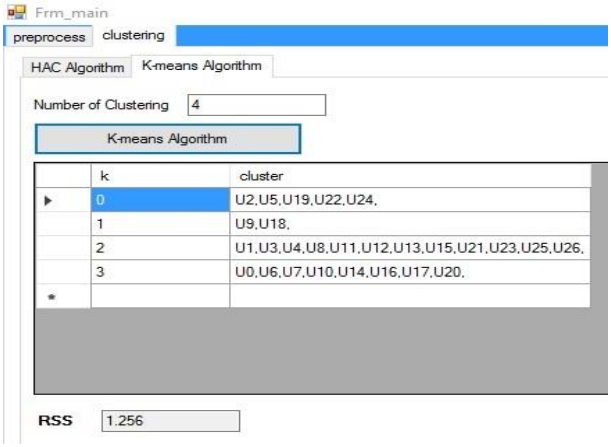


Fig. 9: K-Mean Clustering when K=4.

In case of using k=5, the result is shown in Fig.10. It is observed that all of user divided into five clusters.

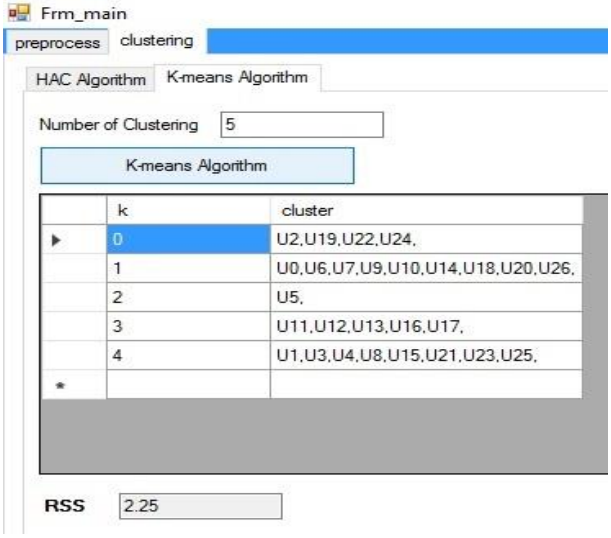


Fig. 10: K-Mean Clustering when K=5.

In case of using k=6, the result is shown in Fig.11. It is observed that all of user divided into six clusters.

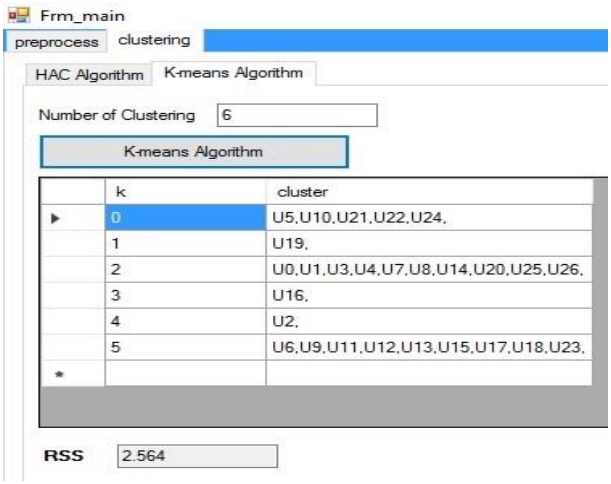


Fig. 11: K-Mean Clustering when K=6.

In case of using k=7, the result is shown in Fig.12. It is observed that all of user divided into seven clusters.

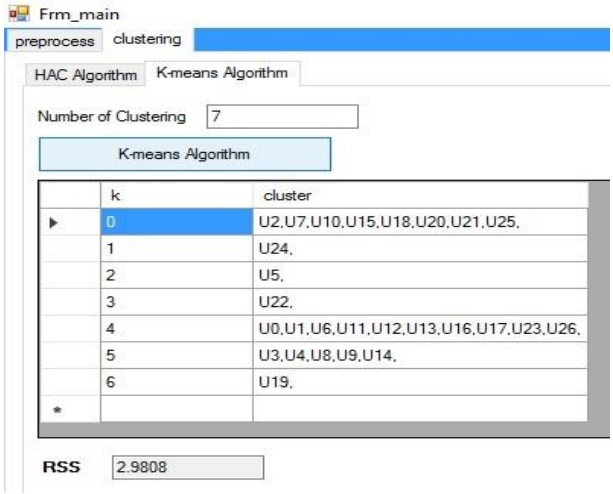


Fig. 12: K-Mean Clustering when K=7.

In case of using k=8, the result is shown in Fig.13. It is observed that all of user divided into eight clusters.

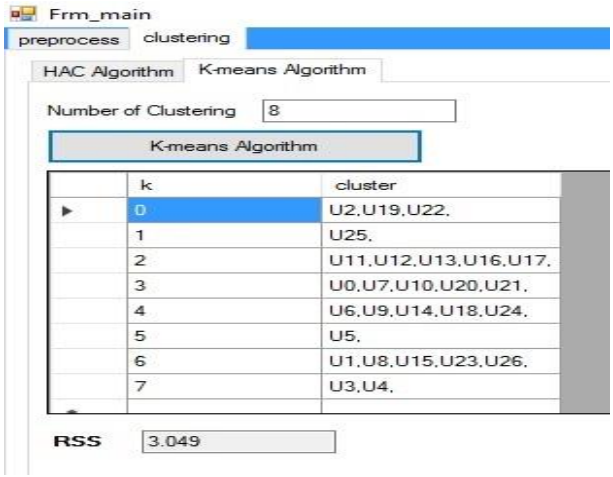


Fig. 13: K-Mean Clustering when K=8.

4.8. Experiment evaluation

In most clustering frameworks, the objective functions try to achieve a high intra-cluster similarity and a low inter-cluster similarity, and this is often considered as an internal clustering quality criterion. One of the measures of internal criterion is the residual sum of squares (RSS), defined as the cosine similarity of each vector from its centroid summed over all vectors[40]. The RSS value is calculated thus:

RSS= $\sum_{k=1}^{k} RSS_k$ where $RSS_k \sum_{x \in w_k} Cosine_Sim(\vec{x}, u(w_k))$ The aim is to maximize the RSS value as it relates to the maximization of the similarity in the same cluster[40].The clustering results are shown in Table 2.

Table 2: Clustering Result

K	RSS
2	0.8722
3	1.1147
4	1.256
5	2.25
6	2.564
7	2.9808
8	3.049

From Table 2, we can see that when the value k is set at 8, the similarity preference of the algorithm was the best, so, the clustering result is shown in Fig.14 based on k = 8.

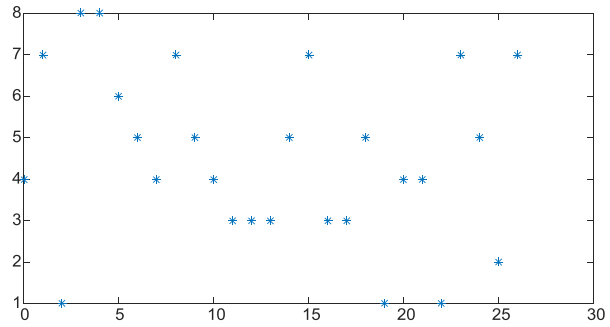


Fig. 14: Users after Clustering.

The relationship between k and RSS is shown in Fig. . The RSS was observed to be highest when the value of k = 8, meaning that the users' similarity in the same cluster is high.

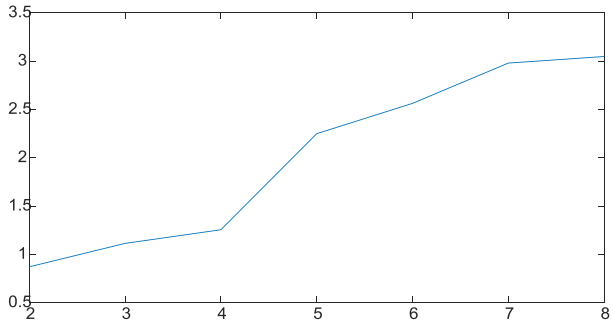


Fig. 15: Relationship between RSS and Number of Clustering.

5. Conclusion

In this paper, we presented the use of the k-means algorithm for the clustering of web users using session-based similarities. The clustering was intended to model the similarities between web users as characterized based on cosine similarity measures. A web user may frequently visit a web page and spend an arbitrary amount of time per visit; users may also access a web page for different reasons. Hence, our web user clustering was based on user sessions rather than the user's entire history. For the data set, the web servers contain 3446 pages and 27 sessions after cleaning and pre-processing. We implemented RSS as a measure of the internal criterion of clustering quality to maximize the similarity in the same clusters. The experiments were conducted, and the results showed that the proposed method can cluster web users with similar interests.

Acknowledgement

This work is supported by the Ministry of Higher Education (MOHE) and the Research Management Centre (RMC) at the Universiti Teknologi Malaysia (UTM) under the Research University Grant Category (VOT Q.J130000.2528.16H74 and R.J130000.7828.4F985).

References

[1] Kettani, O., F. Ramdani and B. Tadili. AK-means: an automatic clustering algorithm based on K-means. Journal of Advanced Computer Science & Technology. 2015. 4(2): 231. https://doi.org/10.14419/jacst.v4i2.4749.

[2] Manukonda, S. R. and N. Divya. Efficient document clustering for web search result. International Journal of Engineering and Technology (UAE). 2018. 7(3): 90-92. https://doi.org/10.14419/ijet.v7i3.3.14494.

[3] Sabitha, V. and D. S.K. Srivatsa. An Efficient Modified K-Means and Artificial Bee Colony Algorithm for Mining Search Result from Web Database. International Journal of Engineering & Technology. 2018. 7(2.20)5.

[4] Silverstone, R. Introduction. Media, technology and everyday life in Europe. Routledge. 19-36; 2017.

[5] Satish Babu, J., T. Ravi Kumar and D. Shahana Bano. Optimizing webpage relevancy using page ranking and content based ranking. 2018. 2018. Seven (2.7) five.

[6] Narayan Jadhav, J. and B. Arunkumar. Web Page Recommendation System Using Laplace Correction Dependent Probability and Chronological Dragonfly-Based Clustering. 2018. 2018. Seven (3.27): 13.

[7] Catledge, L. D. and J. E. Pitkow. Characterizing browsing strategies in the World-Wide Web. Computer Networks and ISDN systems. 1995. 27(6): 1065-1073. https://doi.org/10.1016/0169-7552(95)00043-7.

[8] Shahabi, C., A. M. Zarkesh, J. Adibi and V. Shah. Knowledge discovery from user's web-page navigation. Research Issues in Data Engineering, 1997. Proceedings. Seventh International Workshop on: IEEE. 1997. 20-29.

[9] Yan, T. W., M. Jacobsen, H. Garcia-Molina and U. Dayal. From user access patterns to dynamic hypertext linking. Computer Networks and ISDN Systems. 1996. 28(7): 1007-1014. https://doi.org/10.1016/0169-7552(96)00051-7.

[10] Cunha, C. R. and C. E. Jaccoud. Determining www user's next access and its application to pre-fetching. Computers and Communications, 1997. Proceedings. Second IEEE Symposium on: IEEE. 1997. 6-11.

[11] Cao, P. and S. Irani. Cost-Aware WWW Proxy Caching Algorithms. Usenix symposium on internet technologies and systems. 1997. 193-206.

[12] Cao, P., J. Zhang and K. Beach. Active cache: Caching dynamic contents on the web. Distributed Systems Engineering. 1999. 6(1): 43. https://doi.org/10.1088/0967-1846/6/1/305.

[13] Cooley, R., B. Mobasher and J. Srivastava. Data preparation for mining World Wide Web browsing patterns. Knowledge and information systems. 1999. 1(1): 5-32. https://doi.org/10.1007/BF03325089.

[14] Fu, Y., K. Sandhu and M.-Y. Shih. Clustering of web users based on access patterns. Proceedings of the 1999 KDD Workshop on Web Mining: San Diego, CA. Springer-Verlag. 1999.

[15] Su, Q. and L. Chen. A method for discovering clusters of e-commerce interest patterns using click-stream data. Electronic commerce research and applications. 2015. 14(1): 1-13. https://doi.org/10.1016/j.elerap.2014.10.002.

[16] Yuvaraj, K. and D. Manjula. A performance analysis of clustering based algorithms for the microarray gene expression data. International Journal of Engineering and Technology (UAE). 2018. 7(2): 201-205. https://doi.org/10.14419/ijet.v7i2.21.12172.

[17] Aparajita, A., S. Swagatika and D. Singh. Comparative analysis of clustering techniques in cloud for effective load balancing. International Journal of Engineering and Technology (UAE). 2018. 7(3): 47-51. https://doi.org/10.14419/ijet.v7i3.4.14674.

[18] Patil, H. and R. Singh Thakur. A semantic approach for text document clustering using frequent itemsets and WordNet. 2018. 2018.7 (2.9) four.

[19] Srivastava, J., R. Cooley, M. Deshpande and P.-N. Tan. Web usage mining: Discovery and applications of usage patterns from web data. ACM SIGKDD Explorations Newsletter. 2000. 1(2): 12-23. https://doi.org/10.1145/846183.846188.

[20] Mobasher, B., H. Dai, T. Luo and M. Nakagawa. Effective personalization based on association rule discovery from web usage data. Proceedings of the third international workshop on Web information and data management: ACM. 2001. 9-15. https://doi.org/10.1145/502932.502935.

[21] Yang, Q., H. H. Zhang and T. Li. Mining web logs for prediction models in WWW caching and prefetching. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining: ACM. 2001. 473-478. https://doi.org/10.1145/502512.502584.

[22] Li, I. T. Y., Q. Yang and K. Wang. Classification Pruning for Web-request Prediction. WWW Posters. 2001.

[23] Mobasher, B., R. Cooley and J. Srivastava. Creating adaptive web sites through usage-based clustering of URLs. Knowledge and Data Engineering Exchange, 1999. (KDEX'99) Proceedings. 1999 Workshop on: IEEE. 1999. 19-25.

[24] Pallis, G., L. Angelis and A. Vakali. Model-based cluster analysis for web user is sessions. Foundations of Intelligent Systems. Springer. 219-227; 2005 https://doi.org/10.1007/11425274_23.

[25] Xiao, J. and Y. Zhang. Clustering of web users using session-based similarity measures. Computer Networks and Mobile Computing, 2001. Proceedings. 2001 International Conference on: IEEE. 2001. 223-228.

[26] Xu, J. and H. Liu. Web user clustering analysis based on KMeans algorithm. Information Networking and Automation (ICINA), 2010 International Conference on: IEEE. 2010. V2-6-V2-9.
[27] Chitraa, V. and A. S. Thanamani. An Enhanced Clustering Technique for Web Usage Mining. International Journal of Engineering Research & Technology (IJERT) Vol. 2012. 1.
[28] Poornalatha, G. and P. S. Raghavendra. Web user session clustering using modified K-means algorithm. Advances in Computing and Communications. Springer. 243-252; 2011
[29] Duraiswamy, K. and V. V. Mayil. Similarity matrix based session clustering by sequence alignment using dynamic programming. Computer and Information Science. 2008. 1(3): 66. https://doi.org/10.5539/cis.v1n3p66.
[30] Xiao, J., Y. Zhang, X. Jia and T. Li. Measuring similarity of interests for clustering web-users. Proceedings of the 12th Australasian database conference: IEEE Computer Society. 2001. 107-114.
[31] Sastry, J. K. R., N. Sreenidhi and K. Sasidhar. Quantifying quality of WEB site based on usability. International Journal of Engineering and Technology (UAE). 2018. 7(2.7 Special Issue 7): 320-322.
[32] Romano, S. and H. ElAarag. A neural network proxy cache replacement strategy and its implementation in the Squid proxy server. Neural computing and Applications. 2011. 20(1): 59-78. https://doi.org/10.1007/s00521-010-0442-0.
[33] Jadhav, J. N. and B. Arunkumar. Web page recommendation system using laplace correction dependent probability and Chronological dragonfly-based clustering. International Journal of Engineering and Technology (UAE). 2018. 7(3.27 Special Issue 27): 290-302.
[34] Kaplan, A. M. and M. Haenlein. Users of the world, unite! The challenges and opportunities of Social Media. Business horizons. 2010. 53(1): 59-68. https://doi.org/10.1016/j.bushor.2009.09.003.
[35] Consortium, W. W. W. RDF 1.1 concepts and abstract syntax. 2014.
[36] NLANR, M. B., National Laboratory for Applied Network Research. 2006.
[37] Abhari, A., S. P. Dandamudi and S. Majumdar. Web object-based storage management in proxy caches. Future Generation Computer Systems. 2006. 22(1-2): 16-31. https://doi.org/10.1016/j.future.2005.08.003.
[38] Jain, A. K. Data clustering 50 years beyond K-means. Pattern recognition letters. 2010. 31(8): 651-666. https://doi.org/10.1016/j.patrec.2009.09.011.
[39] Lingras, P. and C. West. Interval set cl.ustering of web users with rough k-means. Journal of Intelligent Information Systems. 2004. 23(1): 5-16. https://doi.org/10.1023/B:JIIS.0000029668.88665.1a.
[40] Singh, V. K., N. Tiwari and S. Garg. Document clustering using k-means, heuristic k-means and fuzzy c-means. Computational Intelligence and Communication Networks (CICN), 2011 International Conference on: IEEE. 2011. 297-301.