Cluster Computing (2019) 22:S145–S155
https://doi.org/10.1007/s10586-018-1959-8

Handling high web access utility mining using intelligent hybrid hill climbing algorithm based tree construction

T. Tamilselvi[1] · G. Tholkappia Arasu[1]

Received: 20 November 2017 / Revised: 18 January 2018 / Accepted: 31 January 2018 / Published online: 23 February 2018

Abstract

With an eye on locating the sequences from web data the most talented solution web usage mining is elegantly employed. In this regard, the modern technique is plagued by several constraints like the forward reference of web access sequence, making it irrelevant for the incremental databases. With the intention of overwhelming these hassles, an innovative technique is elegantly launched for mining web access utility by employing the hybrid hill climbing genetic algorithm (HHCGA) based tree construction. The novel approach deploys the two tree constructions such as the HUWAS tree (HHCGA and utility based web access sequence tree) and the HIUWAS tree (HHCGA and incremental utility based web access sequence tree). The tree construction, in essence, is generally dependent on the internal and external utility values of the web access sequence. The HHCGA is effectively utilized to optimize both the HUWAS and HIUWAS trees. In this regard, hill climbing attracts attention as an optimization approach offering solutions for the search challenges and Genetic Algorithm makes its presence felt as an ideal one for issues encompassing an extensive and intricate search space with added local optimums. The epoch-making technique comes out with flying colors by effectively addressing both the forward and backward references of web access.

Keywords Web access sequence · Web data · Web usage mining · Intelligent hill climbing · Genetic algorithm · Local optimum · Mining utility

1 Introduction

Web mining is the process of extracting information and patterns from web [13]. Web mining can be categorized into three different classes based on which part of the Web is to be mined. These three categories are (i) Web content mining, (ii) Web structure mining and (iii) Web usage mining [21]. Web usage mining is defined as the extraction of meaningful user patterns from web server access logs using data mining techniques [7]. The term web usage mining was introduced by Cooley et al. in 1997 and in accordance with their definition; web usage mining is the automatic discovery of user access patterns from web servers [17, 22]. The goal of Web usage mining is to capture, model, and analyze the behavioral patterns and profiles of users interacting with a Web site [6]. Web Usage Mining deals with understanding of user behavior, while interacting with web site, by using various log files to extract knowledge from them [19]. Details like user log files, request for resources etc., are maintain in web servers, which is the core mining area of web usage. The analysis of these gives the user browsing patterns and that can be utilized for target advertisement, enhancement of web design, satisfaction of customers and making market analysis. Most of the e-service providers realized the fact that they can apply this tool to retain their customers [12].

As every data mining task, the process of Web usage mining also consists of three main steps: (i) pre-processing, (ii) pattern discovery and (iii) pattern analysis [21, 22]. Preprocessing is an important step because of the complex nature of the Web architecture which takes 80% in mining process [9]. This step cleans and filters the web log data and identifies the sessions from it [3]. In the task of data preparation server session file build, where each session is

✉ T. Tamilselvi
tamilselvime@gmail.com

G. Tholkappia Arasu
tholsg@gmail.com

[1] AVS College of Technology, Salem, India

a sequence of requests of different types made by single user during a single visit to a site. In the pattern discovery task association rules, sequential patterns, usage clusters, page clusters, user classification involve [13]. The third step is pattern analysis in which tools are provided to facilitate the transformation of information into knowledge.

Knowledge query mechanism such as SQL is the most common method of pattern analysis [9].

Web usage mining is the application of sequential pattern mining techniques to discover usage patterns from Web data, in order to understand and better serve the needs of Web-based applications [10]. The use of sequence mining techniques for web usage mining is increasingly popular [17]. Sequential pattern mining has emerged as an important topic in data mining. It has proven to be very essential for handling order-based critical business problems, such as behavior analysis, gene analysis in bioinformatics and weblog mining [23, 22]. The purpose of sequence mining is to find frequent sequences that exceed a user-specified support threshold [5, 10]. A sequential pattern mining algorithm mines the sequence database looking for repeating patterns (known as frequent sequences) that can be used later by end users or management to find associations between the different items or events in their data for purposes such as marketing campaigns, business reorganization, prediction and planning [20]. Sequential pattern mining is an important data mining problem with broad applications, including the analyses of customer purchase behavior, web access patterns, scientific experiments, disease treatments, natural disaster and protein formations [10].

2 Related works

Several techniques were proposed by various authors for web usage mining and a few of them are explained below:

Nasraoui et al. [8] have proposed a complete framework and findings in mining Web usage pattern from Web log files of a real Website that had all the challenging aspects of real-life Web usage mining, including evolving user profiles and external data describing an ontology of the Web content. They have presented an approach for discovering and tracking evolving user profiles. They have also described how the discovered user profiles could be enriched with explicit information need that was inferred from search queries extracted from Web log data. Profiles were also enriched with other domain-specific information facets that given a panoramic view of the discovered mass usage modes. An objective validation strategy was also used to assess the quality of the mined profiles, in particular their adaptability in the face of evolving user behavior.

Ahmed et al. [1] have proposed a three novel tree structures to efficiently perform incremental and interactive HUP mining. The first tree structure, incremental HUP lexicographic tree (IHUPL-Tree), was arranged according to an item's lexicographic order. It could capture the incremental data without any restructuring operation. The second tree structure was the IHUP transaction frequency tree (IHUPTF-Tree), which obtained a compact size by arranging items according to their transaction frequency (descending order). To reduce the mining time, the third tree, IHUP-transaction-weighted utilization tree (IHUPTWU-Tree) was designed based on the TWU value of items in descending order. Extensive performance analyses have shown that our tree structures were very efficient and scalable for incremental and interactive HUP mining.

Pierrakos et al. [11] have proposed a knowledge discovery framework for the construction of Community Web Directories. They have introduced a concept on applying personalization to Web directories. The Web directory was viewed as a thematic hierarchy and personalization was realized by constructing user community models on the basis of usage data. They have introduced a methodology that combines the users' browsing behavior with thematic information from the Web directories. Following this methodology, they have also enhanced that the clustering and probabilistic approaches presented in previous work and also present a new algorithm that combines these two approaches. The resulting community models have taken the form of Community Web Directories. The proposed personalization methodology was evaluated both on a specialized artificial and a general-purpose Web directory, indicating its potential value to the Web user.

Tiwari et al. [15] have proposed a web mining solution to business intelligence to discover hidden patterns and business strategies from their customer and web data. Recommender systems have emerged as powerful tools for helping customers find items of interest. They have proposed a new framework based on web mining technology for structure a Web-page recommender system. They have also demonstrated how web mining technology could be effectively applied in a business intelligence environment. Web mining has attempt to determine useful knowledge from secondary data obtained from the interactions of the users with the web.

Fong et al. [4] have proposed a semantic web usage mining approach for discovering periodic web access patterns from annotated web usage logs which incorporates information on consumer emotions and behaviors through self-reporting and behavioral tracking. Here they have used a fuzzy logic to represent real-life temporal concepts and requested resource attributes of periodic pattern based web access activities. Those fuzzy temporal and resource

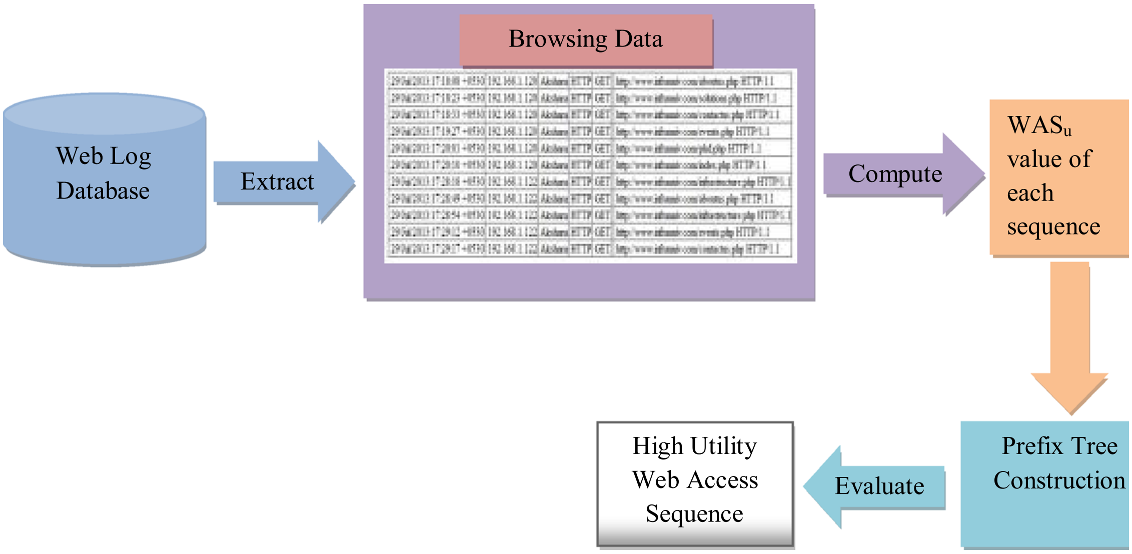


Fig. 1 Block diagram of proposed method

representations, which contain both behavioral and emotional cues, were incorporated into a Personal Web Usage Lattice that models the user's web access activities. From that, they have generated a Personal Web Usage Ontology written in OWL, which enabled semantic web applications such as personalized web resources recommendation. Finally, they have also demonstrated the effectiveness of their approach by presenting experimental results in the context of personalized web resources recommendation with varying degrees of emotional influence. Emotional influence has been found to contribute positively to adaptation in personalized recommendation.

Varghese et al. [18] have proposed a cluster optimization methodology based on fuzzy logic and that was used for eliminating the redundancies occur in data after clustering done by web usage mining methods. For clustering Fuzzy C-Means algorithm was used. Fuzzy Cluster-chase algorithm for cluster optimization was presented to personalize web page clusters of end users. Here Web log file was given as input and perform data cleaning to eliminate irrelevant data items. The cleaned web log was used for pattern discovery. Clustering techniques was used for discovering useful usage patterns.

Mining high utility item sets from a transactional database refers to the discovery of item sets with high utility like profits. Tseng et al. [2, 16] have proposed two algorithms, namely utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility item sets with a set of effective strategies for pruning candidate item sets. The information of high utility item sets was maintained in a tree-based data structure named utility pattern tree (UP-

Fig. 2 Header table for HUWAS tree

Header Table
G_1=529
G_2=483
G_3=387
G_4=46
G_5=308
G_6=102

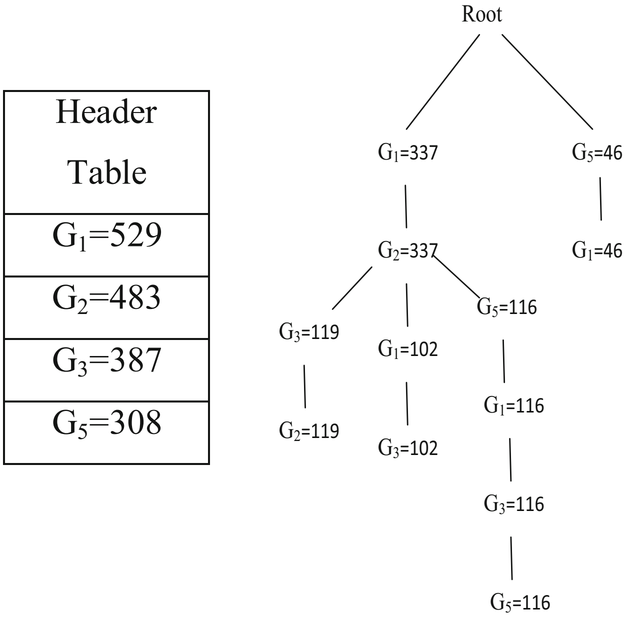


Fig. 3 HUWAS tree construction

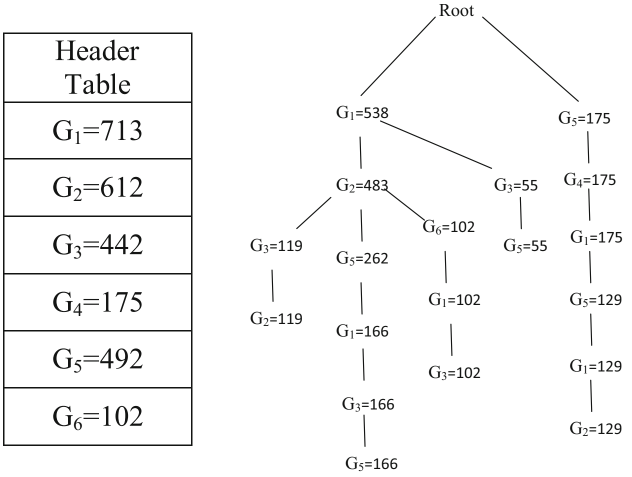


Fig. 4 HIUWAS Tree (after inserting WASDb*)

Tree) such that candidate item sets could be generated efficiently with only two scans of database. The performance of UP-Growth and UP-Growth + was compared with the state-of-the-art algorithms on many types of both real and synthetic data sets. Experimental results have shown that the proposed algorithms, especially UP-Growth+, not only reduce the number of candidates effectively but also outperform other algorithms substantially in terms of runtime, especially when databases contain lots of long transactions.

3 Problem definition

Web usage mining is the automatic detection of user access patterns from the servers on the web. The companies regain vast quantities of data from their day to day actions which are generally created by the web servers and are kept in the server access logs. The problems in the existing web usage mining process are given below.

The web log could be on the server side, client-side or on a proxy server, each having its own benefits and drawbacks on finding the users' relevant patterns and navigational sessions.

The majority of off-line procedure have the drawback on reduction of accuracy over time resulted of new users joining or modifications of pattern for present users in model-based techniques.

In the initial solution of frequent pattern mining in web usage mining, test paradigm and candidate set generation of Apriority has shown many drawbacks including that it requires multiple database scans and it generates many candidate item sets.

The problems of sequential pattern in web usage mining is that of finding the set of all frequent sequences in the given sequence database of items.

Table 1 An example of web access sequence database with internal utility

Sequence	WAS with I_u	WAS_u
Original $WASD_b$		
q_1	$G_1(4)G_2(5)\ G_6(3)\ G_1(3)\ G_3(6)$	102
q_2	$G_5(3)\ G_4(3)\ G_1(4)$	46
q_3	$G_1(3)G_2(6)\ G_5(3)\ G_1(7)\ G_3(4)\ G_5(5)$	166
q_4	$G_1(5)\ G_2(8)\ G_3(3)\ h(4)$	119
$WASD_b^*$		
q_5	$G_1(4)\ G_2(8)\ G_5(3)$	96
q_6	$G_5(4)G_4(7)\ G_1(8)\ G_5(3)\ G_1(2)\ G_2(3)$	129
q_7	$G_1(1)\ G_3(7)\ G_5(2)$	55

Table 2 An example of web pages with external utility

Web page	E_u
G_1	4
G_2	7
G_3	5
G_4	2
G_5	8
G_6	3

The worst thing that's a threat in web usage mining, the invasion of privacy. Privacy is generally considered to be lost when the documents of a person obtained, broadcast or used mainly when it occurs without the presence of the person who came up with the data itself. Companies for various reasons and data collection purposes.

In the existing work [11] for personalizing web directories with the aid of web usage, the Data Proxy server logs have making a number of challenges, such as the handling of their size and semantic diversity. Knowledge discovery methods cannot be adapted to the task of discovering community directories. Also the other classification methods cannot be exploited for the initial mapping of the Web pages to the Web directory.

A web usage mining framework for Business Intelligence was an existing work in Ref. [15]. This paper makes several contributions to the framework of recommender systems linked research. Research for analyzing customers past purchasing pattern cannot enable to discover an appropriate. Business intelligence framework used a visual web log mining architecture so the web mining and visualization of web services log data was not captured in business intelligence environment.

These are the problems of existing works and which motivate us to do this research in the field of web usage mining.

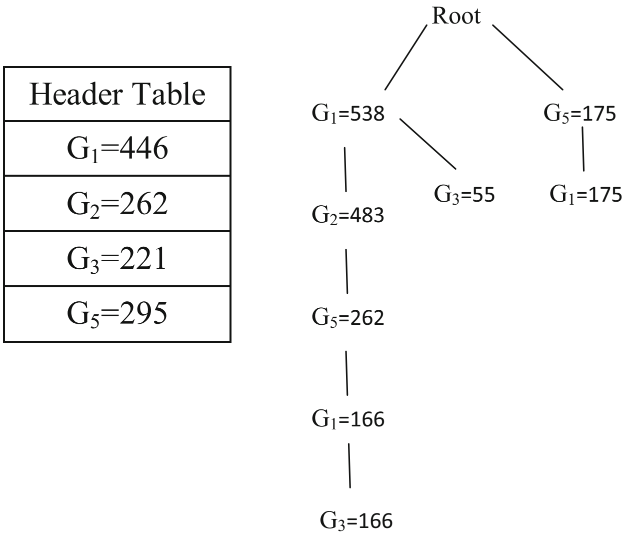


Fig. 5 Conditional tree for web page G_5

4 Proposed method

With the motive of successfully tackling the web access sequences, we have come out with a thriving web utility mining system endowed with elevated utility web access. The most vital intricacy addressed by the novel technique is to duly navigate the web sequence in both the forward and backward references. To accomplish this, we have envisioned two tree constructions in accordance with the hybrid hill climbing genetic algorithm (HHCGA), which are competent to accomplish on the static and dynamic sequences of web access as detailed below:

HUWAS tree (HHCGA and utility based web access sequence tree).

HIUWAS tree (HHCGA and incremental utility based web access sequence tree).

In the case of a user-defined threshold the HUWAS tree construction is capable of achieving the web access sequence data with a maximum of three database scans in a tremendously compact structure. Further, the HIUWAS tree is competent to execute the incremental mining adeptly with a maximum of only two database scans for mining all the consequential high utility web access sequences. The entire block diagram of the innovative mining web access utility is elegantly exhibited in the following Fig. 1.

4.1 A structure for utility web access sequence

Let X = {x_1,x_2,….x_m} represent a set of web pages and $WASD_b$ signify a web access sequence database consisting of a number of sequence such as the {q_1,q_2,….q_k}. At this juncture each sequence q_i of database $WASD_b$($1 \leq i \leq k$) contains a list of web pages x_1,x_2,….x_m($x_j \varepsilon X$, $1 \leq j \leq m$).

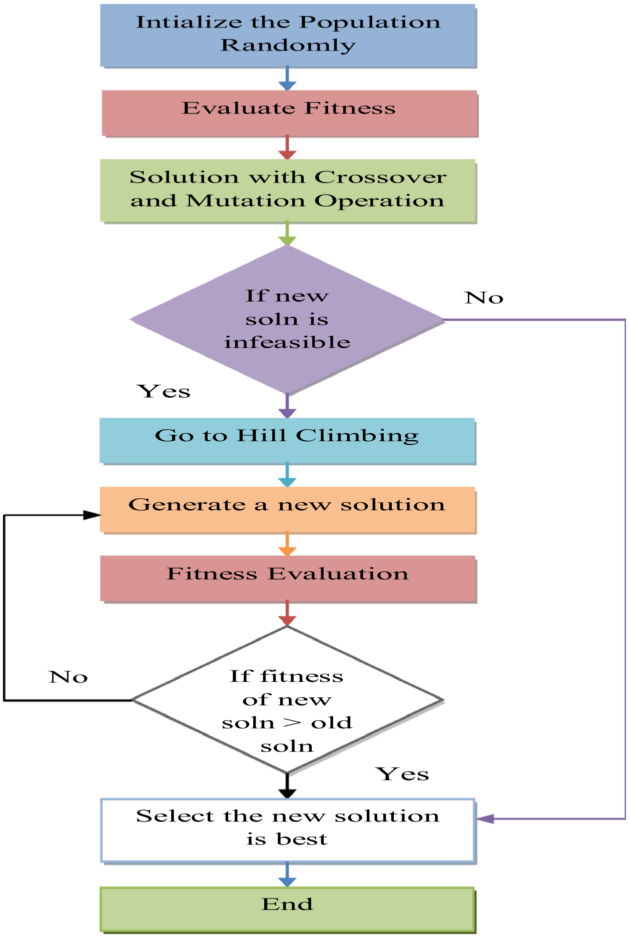

Intialize the Population Randomly
Evaluate Fitness
Solution with Crossover and Mutation Operation
If new soln is infeasible
No
Yes
Go to Hill Climbing
Generate a new solution
Fitness Evaluation
If fitness of new soln > old soln
No
Yes
Select the new solution is best
End

Fig. 6 Flow chart for HHCGA

The length $|q_i|$ of an access sequence indicates the total number of web pages q_i.

The internal–external and utility values of the web pages x_j in the web access sequence q_i are represented by $I_u(x_j,q_i)$ and $E_u(x_j)$ respectively. The web access sequence utility $was_u(x_j,q_i)$ symbolizes the quantitative measure of utility for web page x_j in sequence q_i is expressed by the relation 1 shown below:

$$was_u(x_j, q_i) = I_u(x_j, q_i) \times E_u(x_j) \tag{1}$$

Web access sequence utility of a WAS sequence is furnished by the following Eq. 2,

$$was_u(q_i) = \sum_{x_j \in q_i} was_u(x_j, q_i) \tag{2}$$

Web access sequence utility of a WAS database is given by the formula 3 shown as follows,

$$was_u(WASD_b) = \sum_{q_i \in WASD_b} was_u(q_i) \tag{3}$$

Minimum WAS utility threshold (T) is represented by the percentage of the WAS utility value of the database and is expressed by the relation 4 shown here under.

$$mWAS_u = T \times was_u(WASD_b) \tag{4}$$

An example of web access sequence database with internal and external utilities is elegantly exhibited as following Tables 1 and 2:

In our innovative model, the Internal utility of G_6 in a sequence q_1 represented by $I_u(G_6,q_1) = 3$. Conversely, a web page becomes visible a number times in a web access sequence and in that case $I_u(x_j,q_i)$ representing the sum total of all the quantities of x_j in q_i $I_u(G_1,q_1) = 7$. Subsequently, the external utility of G_1 in a sequence q_1 is $E_u(G_1) = 4$. In the novel technique, we have generated the random number for internal and external utilities in the range of (1–10) and (1.0–10.0) respectively. Subsequently, the web access sequence utility is evaluated by means of the product of internal and external utility as shown below:

$$was_u(G_1, q_1) = I_u(G_1, q_1) \times E_u(G_1) = 7 \times 4 = 28$$

An identical evaluation is carried out the entire web pages in the each sequence. Thereafter, the Web access sequence utility of a WAS q_i is evaluated. Let us take for instance,

$$was_u(WASD_b) = \sum_{q_i \in WASD_b} was_u(q_i)$$

$$\begin{aligned} was_u(q_1) &= was_u(G_1,\ q_1) \\ &\quad + was_u(G_2,\ q_1) + was_u(G_6,\ q_1) \\ &\quad + was_u(G_3,\ q_1) \\ &= 28 + 35 + 9 + 30 = 102 \end{aligned}$$

Subsequently, the was_u of web access sequence database is estimated as follows:$WASD_b$ = original database + $WASD_b^*$

$$was_u(WASD_b) = \sum_{q_i \in WASD_b} was_u(q_i)$$

$$\begin{aligned} Was_u(WASD_b) &= 102 + 46 + 166 + 119 + 96 + 129 + 55 \\ &= 713 \end{aligned}$$

Thereafter, the minimum WAS utility based on the threshold value is evaluated, and in our technique, the threshold value is chosen as 0.25.

$$\begin{aligned} mWAS_u &= T \times Was_u(WASD_b) \\ &= 0.25 \times 713 \\ &= 178.25 \end{aligned}$$

The above work-outs relate to the innovative framework for our performed technique.

4.2 Tree structure for implemented method

4.2.1 HUWAS tree

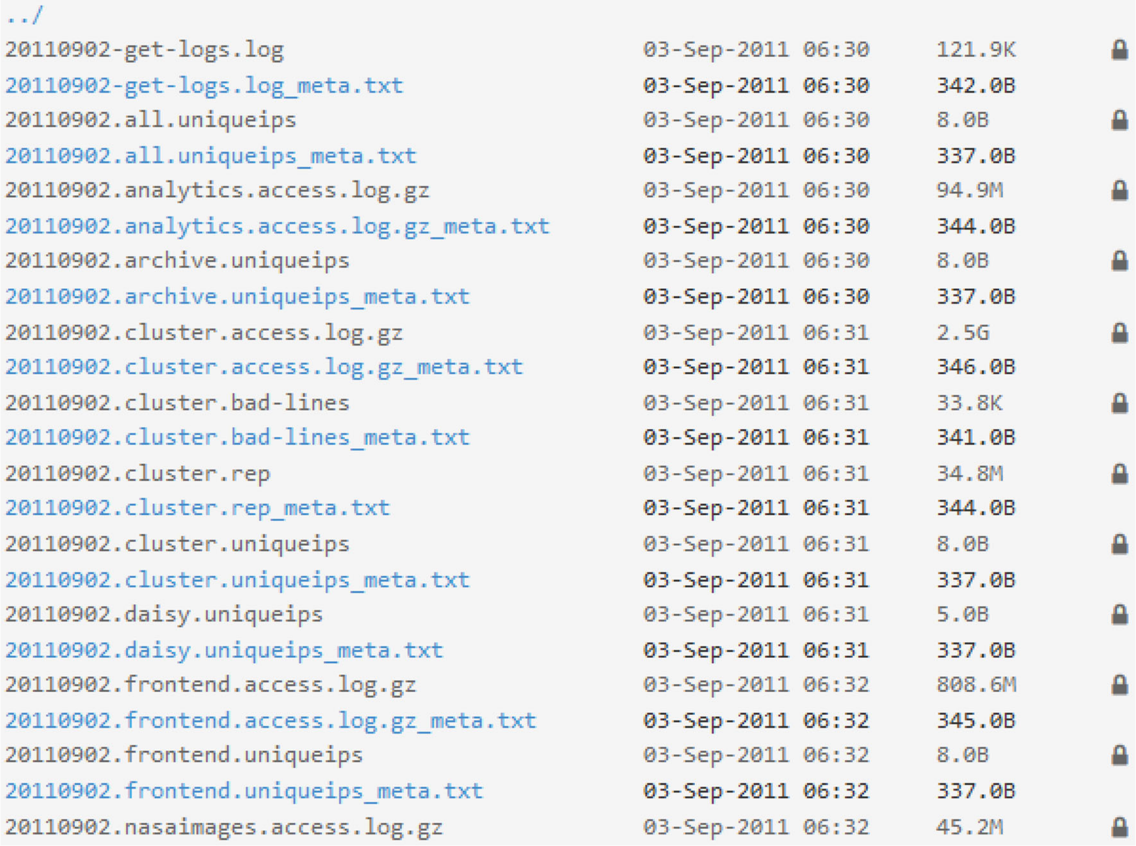
At the outset, we set out to define precisely the construction procedure of our performed tree structure HUWAS tree for mining the high utility web access sequence in the static databases. HUWAS tree structure invariably captures the data and user-defined minimum threshold (T). In the HUWAS tree, preliminary scan identifies the candidate web pages and in the subsequent data base scan, candidate web pages are selected from every WAS and integrated into a HUWAS tree. From the candidate sequence a third data base scan effectively locates the high utility web access sequence. In the HUWAS tree framework, the header table is maintained and each entry of header table preserves both the item id and was_u values for each and every web page. The HUWAS tree header table is elegantly exhibited in Fig. 2.

In the original database the innovative HUWAS tree has the minimum WAS utility based on the threshold value (T = 0.25). Here $mWAS_u$ value of the original database is 132.25. The proposed header table illustrates the was_u values for the web pages as $< G_1 = 529,\ G_2 = 483,\ G_3 = 387,\ G_4 = 46,\ G_5 = 308,\ G_6 = 102 >$. The web pages G_4 and G_6 are not candidates as they tend to smaller than the minimum WAS utility (G_4 and $G_6 < mWAS_u$). Therefore, the novel HUWAS tree has the candidates which are $< G_1,\ G_2,\ G_3,\ G_5 >$. Header table with HUWAS tree construction is effectively demonstrated in Fig. 3. HUWAS tree consists of a number of sequences in the original database.

It is not possible to utilize the innovative HUWAS tree if the database is enhanced. As Fig. 3 is stepped up by adding two web access sequence with the identical user threshold value (T = 0.25), the candidate web page G_4 emerges as a candidate sequence though initially he HUWAS tree web page G_4 was not a candidate. If the database is stepped up, it has to be configured right from the initial phase. In our technique, we have employed the HIUWAS tree construction to successfully address the challenges.

4.2.2 HIUWAS tree

At this junction, we clearly define the HIUWAS tree construction (incremental HHCGA utility based web access sequence tree), which is not dependent on the user-defined threshold value and it captures the whole web access sequence database within their was_u values. The following Fig. 4 clearly illustrates the HIUWAS tree construction.



```
../
20110902-get-logs.log                      03-Sep-2011 06:30    121.9K    🔒
20110902-get-logs.log_meta.txt             03-Sep-2011 06:30    342.0B
20110902.all.uniqueips                     03-Sep-2011 06:30    8.0B      🔒
20110902.all.uniqueips_meta.txt            03-Sep-2011 06:30    337.0B
20110902.analytics.access.log.gz           03-Sep-2011 06:30    94.9M     🔒
20110902.analytics.access.log.gz_meta.txt  03-Sep-2011 06:30    344.0B
20110902.archive.uniqueips                 03-Sep-2011 06:30    8.0B      🔒
20110902.archive.uniqueips_meta.txt        03-Sep-2011 06:30    337.0B
20110902.cluster.access.log.gz             03-Sep-2011 06:31    2.5G      🔒
20110902.cluster.access.log.gz_meta.txt    03-Sep-2011 06:31    346.0B
20110902.cluster.bad-lines                 03-Sep-2011 06:31    33.8K     🔒
20110902.cluster.bad-lines_meta.txt        03-Sep-2011 06:31    341.0B
20110902.cluster.rep                       03-Sep-2011 06:31    34.8M     🔒
20110902.cluster.rep_meta.txt              03-Sep-2011 06:31    344.0B
20110902.cluster.uniqueips                 03-Sep-2011 06:31    8.0B      🔒
20110902.cluster.uniqueips_meta.txt        03-Sep-2011 06:31    337.0B
20110902.daisy.uniqueips                   03-Sep-2011 06:31    5.0B      🔒
20110902.daisy.uniqueips_meta.txt          03-Sep-2011 06:31    337.0B
20110902.frontend.access.log.gz            03-Sep-2011 06:32    808.6M    🔒
20110902.frontend.access.log.gz_meta.txt   03-Sep-2011 06:32    345.0B
20110902.frontend.uniqueips                03-Sep-2011 06:32    8.0B      🔒
20110902.frontend.uniqueips_meta.txt       03-Sep-2011 06:32    337.0B
20110902.nasaimages.access.log.gz          03-Sep-2011 06:32    45.2M     🔒
```

Fig. 7 Sample web log file

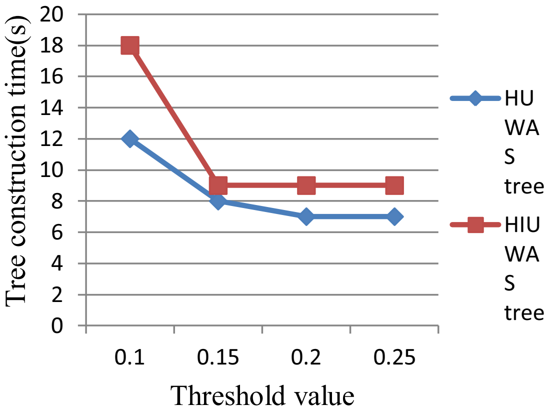
Figure 4a illustrates the HIUWAS tree capturing the original database with was_u value whereas Fig. 4b exhibits the incremental database with was_u value. HIUWAS tree performs a single database scan for ascertaining the high utility web access sequence from the candidate sequence.

4.3 Mining process for implemented method

In this section, a brief account is furnished regarding the mining function of our performed technique, which begins right from the bottom most items. The header table contains the bottom most item G_6 having the minimum value in relation to that of the $mWAS_u$ value. Hence we tend to ignore the web page G_6 and proceed to commence the mining task from the web page G_5. At the out, we consider the entire prefixing paths of web page G_5 with their was_u value. If any sub sequence happens multiple times in an identical path then we reduce the redundancy of was_u value to that path. Now the prefixing paths of G_5 are $< G_{1}$-$G_{2=}262$, $G_1G_2G_5G_1G_3 = 166$, $G_1G_2 = - 166$, $G_5G_4G_1 = 129$, $G_1G_3 = 55 >$. In the innovative technique, the web page G_5 happens two times in the path $G_1G_2G_5G_1G_3$, and the was_u value of G_1G_2 crops up twice with the quality 166. Hence, from $G_1G_2G_5G_1G_3$ we have to

Table 3 Tree construction time for both HUWAS and HIUWAS tree

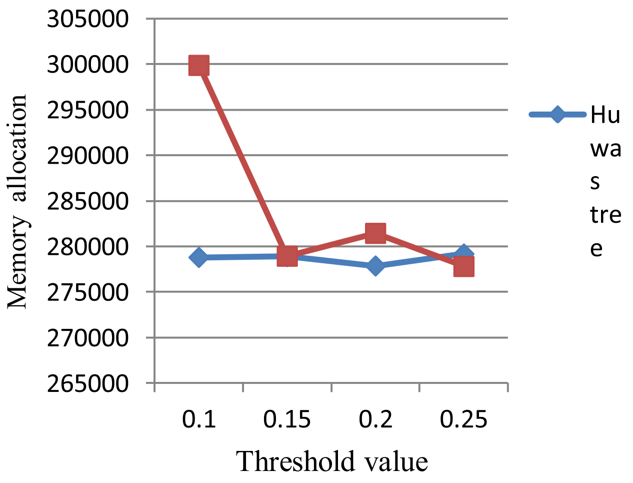
Threshold value	Tree construction time(s)	
	HUWAS tree	HIUWAS tree
0.1	12	18
0.15	8	9
0.2	7	9
0.25	7	9



Graph 1 Tree construction time versus threshold value

Table 4 Memory allocation for both HUWAS and HIUWAS tree

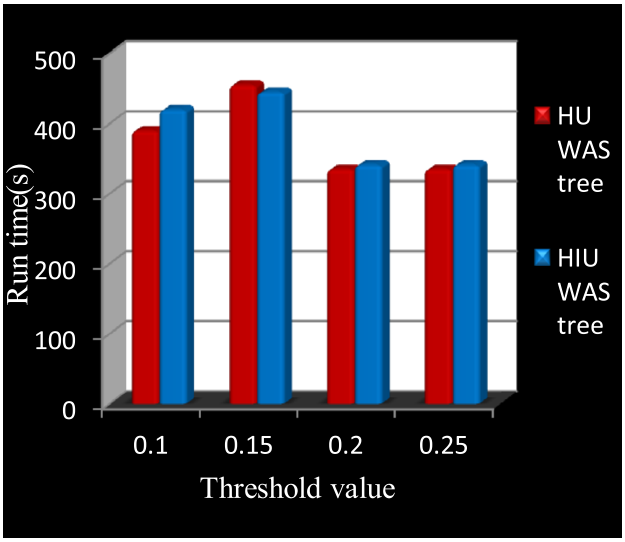
Threshold value	Memory allocation	
	HUWAS tree	HIUWAS tree
0.1	278,808	299,874
0.15	278,896	278,945
0.2	277,872	281,451
0.25	279,184	277,818



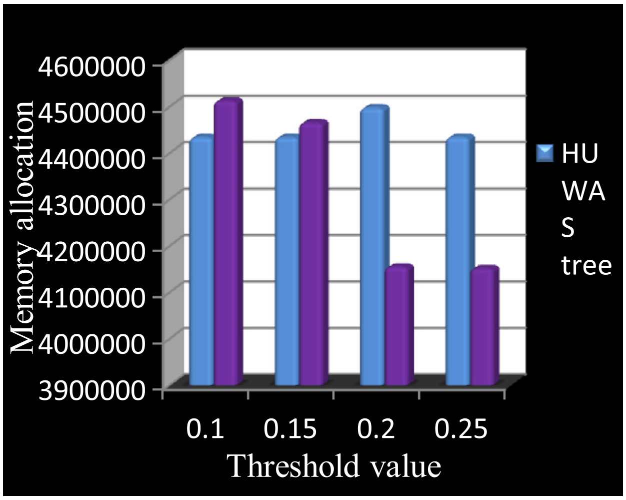
Graph 2 Memory allocation for both HUWAS and HIUWAS tree

subtract the sub sequence G_1G_2 with the was_u value 166. The was_u values of web page G_5 are $< G_1 = 446, G_2 = 262, G_3 = 221, G_4 = 129, G_5 = 295 >$. Now, the web page G_4 does not function as a candidate as $G_4 < mWAS_u$ and the conditional tree for web page G_5 is produced as illustrated in Fig. 5 appearing hereunder.

Subsequently, we create the candidate sequences such as $< G_1G_5 = 446, G_2G_5 = 262, G_3G_5 = 221, G_5G_5 = 295, G_5 = 492 >$. In an identical manner, the candidate sequence prefixing other web pages is evaluated.



Graph 3 Run time for both trees



Graph 4 Memory allocation for both trees

4.4 Hybrid hill climbing genetic algorithm (HHCGA)

The proposed method is producing the enhanced solution by means of HHCGA here Genetic algorithm (GA) employs the result of Hill climbing (HC). Initially, produce the first solution and assess the fitness value. If the fitness

Table 5 Run time and memory allocation for both trees

Threshold value	Run time		Memory allocation	
	HUWAS tree	HIUWAS tree	HUWAS tree	HIUWAS tree
0.1	390	420	4,433,584	4,512,445
0.15	455	445	4,433,672	4,465,114
0.2	335	341	4,497,208	4,154,815
0.25	335	341	4,434,016	4,151,457

Table 6 Mining process execution time and memory allocation for both trees

Threshold value	Execution time		Memory allocation	
	HUWAS tree	HIUWAS tree	HUWAS tree	HIUWAS tree
0.1	425	414	2,979,184	2,784,584
0.15	481	497	3,071,856	3,124,641
0.2	465	455	3,043,440	2,987,541
0.25	457	455	3,404,984	2,986,542

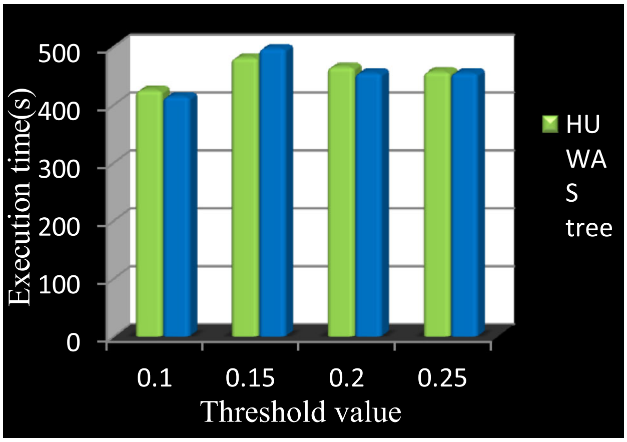
value is lesser than the threshold means that value is called infeasible solution. After applying crossover and mutation operators the new solution is produced in Genetic algorithm. If the new solution is an infeasible solution, it exceeds to a Hill climbing function. For finding local search Hill climbing is an optimization method. Hill climbing begins with infeasible solution next the solution is changed. If the mutation result has higher fitness for the new solution than the old one, the new solution is performed; or else the current solution is retained. Now, the Hill climbing algorithm begins with an infeasible solution and prolongs till a feasible solution is finding next it comes back the feasible solution to Genetic algorithm. The general procedure of HHCGA is elucidated beneath,

Step 1 Generate the initial solution S_i randomly (where $i = 1,2,\ldots,n$)
Step 2 Assess the fitness function

$$fitness = sum\ of\ the\ total\ weight\ for\ each\ user \qquad (5)$$

The fitness value of each parameter is estimated and the greatest fitness value is shortlisted as the best chromosome.

Step 3 Relate the mutation and cross over to the best solution
Mutation In mutation process chromosome values are varied according to the probability
Cross over One or more parent chromosomes are chosen and novel solution is produced after mutation
Step 4 If the new solution is infeasible after that Hill climbing algorithm is performed
Step 5 Enlarge the current solution
Step 6 Discover the fitness function
Step 7 If the fitness value of new solution is greater than the current solution
Step 8 Choose the new solution is the best one



Graph 5 Execution time for the mining process

In Fig. 6 the flowchart for the suggested HHCGA is revealed

5 Results and discussion

In this section, the outcomes of the innovative technique are colorfully carved out. The novel mechanism is executed in the JAVA platform with the system configuration as i5 processor with 4 GB RAM.

5.1 Dataset description

The innovative technique effectively employs the authentic dataset for the investigation. To supervise the web usage of the user 'CC Proxy' software is set up on the server. The route of all users from the website http://www.infrauniv.com/ is efficiently examined. Figure 7 illustrates the sample web log file hereunder.

5.2 Performance analysis

The efficiency in execution of our innovative technique is evaluated in accordance with the tree construction time, run time and memory allocation for both the HUWAS and HIUWAS trees by modifying the threshold value. A detailed account of the entire procedure is elucidated in the ensuing section.

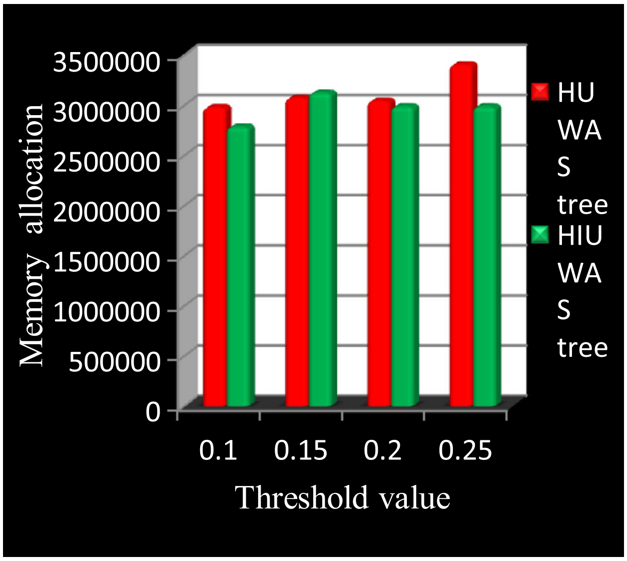
In the novel technique, time consumed for the tree construction is assessed by modifying the threshold value. When the threshold value is 0.1, the time consumed to configure the tree is found to be 12 s for HUWAS tree where as for the HIUWAS tree, the corresponding time is

18 s. When threshold value is set to 0.15, the tree construction time values for the HUWAS and HIUWAS trees are found to be 8 and 9 s respectively. When the threshold is 0.2, HUWAS and HIUWAS tree construction time is 7 and 9 s respectively. When the threshold value is modified to 0.25, the relative time values for the tree construction are found to be 7 and 9 s for HUWAS HIUWAS trees respectively. A clear picture is visible in Table 3 and the values are exhibited in Graph 1 which is illustrated below,

The memory allocation of HUWAS and HIUWAS tree in the innovative technique is exhibited in Table 4, which is carried out by modifying the threshold value in Graph 2.

The innovative HUWAS and HIUWAS trees employ the internal and external utility values to the web access sequence for each user. After integrating the internal and external utility value to the web access sequence the tree is generated. In accordance with this, the run time is determined for both the HUWAS and HIUWAS trees. The memory allocation of both the trees is modified after integrating the internal and external utility values of the web access sequence. The run time and the memory allocation of both the trees are beautifully included in Table 5. The Run time and memory allocation are effectively pictured in Graphs 3 and 4 which are exhibited hereunder.

Mining process of our proposed method is evaluated based on the time and memory allocation of both HUWAS and HIUWAS tree construction. Table 6 represents the time taken for the mining process and memory allocation for the mining process is tabulated. When the threshold value is set as 0.1, the execution time is 425 s in HUWAS tree and 414 s in HIUWAS tree. Here the execution time of HIUWAS tree is minimum compared to HUWAS tree and the memory allocation for the mining process is less in HIUWAS tree compared to HUWAS tree. Table 6 is shown in below,



Graph 6 Memory allocation for the mining process

6 Conclusion

In this document the mining web access utility using hybrid hill climbing genetic algorithm based tree construction is elegantly launched. The HUWAS and HIUWAS constitute the two vital tree constructions used in the innovative technique so as to successfully address the problems faced by the modern techniques. In the novel technique, the tree construction is in accordance with the internal and external utility values of the web access sequence. Both the tree constructions are optimized by the hybrid hill climbing genetic technique. The cheering outcomes of the innovative approach illustrate the skills of our novel technique to perform the incremental mining of high utility web access sequence.

References

1. Ahmed, C.F., Tanbeer, S.K., Jeong, B.S., Lee, Y.K.: Efficient tree structures for high utility pattern mining in incremental databases. IEEE Trans. Knowl. Data Eng. **21**(12), 1708–1721 (2009)
2. Bhattacharya, S., Dubey, D.: High utility item set mining. Int. J. Emerg. Technol. Adv. Eng. **2**(8), 476–481 (2012)
3. Chordia, B.S., Adhiya, K.P.: Grouping web access sequences using sequence alignment method. Indian J. Comput. Sci. Eng. (IJCSE) **2**(3), 308–314 (2011). **In Proceeding of Bhupendra S Chordia et al.**
4. Fong, A.C.M., Zhou, B., Hui, S., Tang, J., Hong, G.: Generation of personalized ontology based on consumer emotion and behavior analysis. IEEE Trans. Affect. Comput. **3**(2), 152–164 (2012)
5. Jing, L., Keech, M., Chen, W.: Concurrency in web access patterns mining. In: Proceeding of IEEE International Conference on World Academy of Science, Engineering and Technology, vol. 3, pp. 842–851, 2009
6. Langhnoja, S., Barot, M., Mehta, D.: Pre-processing: procedure on web log file for web usage mining. Proc. IEEE Int. J. Emerg. Technol. Adv. Eng. **2**(12), 419–423 (2012)
7. Malik, S.K., Rizvi, S.: Information extraction using web usage mining, web scrapping and semantic annotation. In: Proceeding of IEEE International Conference on Computational Intelligence and Communication Systems, pp. 465–469, 2011
8. Nasraoui, O., Soliman, M., Saka, E., Badia, A., Germain, R.: A web usage mining framework for mining evolving user profiles in dynamic web sites. IEEE Trans. Knowl. Data Eng. **20**(2), 202–215 (2008)
9. Nithya, P., Sumathi, P.: Novel pre-processing technique for web log mining by removing global noise and web robots. In: Proceeding of IEEE International Conference on Computing and Communication Systems, pp. 1–5, 2012
10. Parikh, M., Chaudhari, B., Chand, C.: A comparative study of sequential pattern mining algorithms. Proc. IEEE Int. J. Appl. Innov. Eng. Manag. (IJAIEM) **2**(2), 103–109 (2013)

11. Pierrakos, D., Paliouras, G.: Personalizing web directories with the aid of web usage data. IEEE Trans. Knowl. Data Eng. **22**(9), 1331–1344 (2010)
12. Prasanth, A.: Web usage mining—its application in E-services. Proc. Int. Emerg. Technol. Adv. Eng. **3**(2), 572–576 (2013)
13. Sharma, P., Bhartiya, R.: An efficient algorithm for improved web usage mining. In: Proceeding of IEEE International Conference on Computer Technology and Applications vol. 3(2), pp. 766–769, 2013
14. Soares, C., de Graaf E., Kok, J.N., Kosters, W.A.: Sequence mining on web access logs: a case study. In: Proceeding of International Conference on Information Technology, pp. 1–8, 2007
15. Tiwari, S., Richariya, P., Razdan, D., Tomar, S.: A web usage mining framework for business intelligence. In: Proceeding of IEEE 3rd International Conference on Communication Software and Networks (ICCSN), pp. 731–734, May 2011
16. Tseng, V.S., Shie, B.E., Wu, C.W., Philip, S.Y., Fellow, IEEE Abstract: Efficient algorithms for mining high utility item sets from transactional databases. IEEE Trans. Knowl. Data Eng. **25**(8), 1172–1186 (2013)
17. Tyagi, N.K., Solanki, A.K., Tyagi, S.: An algorithmic approach to data preprocessing in web usage mining. Int. J. Emerg. Technol. Knowl. Manag. **2**(2), 279–283 (2010)
18. Varghese, N.M., John, J.: Cluster optimization for enhanced web usage mining using fuzzy logic. In: Proceeding of IEEE International Conference on Information and Communication Technologies (WICT), pp. 948–952, (2012)
19. Varnagar, C.R., Madhak, N.N., Kodinariya, T.M., Rathod, J.N.: Web usage mining: a review on process, methods and techniques. In: Proceeding of IEEE International Conference on Information Communication and Embedded Systems, pp. 40–46, 2013
20. Vijayalakshmi, S., Mohan, V.: Mining sequential access pattern with low support from large pre-processed web logs. Proc. IEEE Int. J. Comput. Sci. **6**(11), 1293–1300 (2010)
21. Vijayalakshmi, S., Mohan, V., Raja, S.S.: Mining of users' access behavior for frequent sequential pattern from web logs. Int. J. Database Manag. Syst. (IJDMS) (2010). https://doi.org/10.5121/ijdms.2010.2304
22. Xu, C., Chen, S., Cheng, J.: Network user interest pattern mining based on entropy clustering algorithm. In: Proceedings of the 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Xi'an, China, 17–19 Sept 2015, pp. 200–204
23. Yin, J., Zheng, Z., Cao, L.: USpan: an efficient algorithm for mining high utility sequential patterns. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 660–668, Aug 2012

T. Tamilselvi Received her B.E. degree in Periyar University, Tamil Nadu in 2004. She received M.E. degree from Computer Science and Engineering Department at Anna University in 2009. She is currently a Ph.D. student at Information and Communication Engineering Department, Anna University. Her research interests include Web usage mining, Semantic Web services and Data Processing.

G. Tholkappia Arasu is currently Principal of AVS College of Technology, Salem. He has received his Ph.D. degree from Information and Communication Engineering Department, Anna University in 2008. His research interests include Agent based Intelligent Systems, Artificial Intelligence, Web Mining and Sensor Networks.