



Personalized web page recommendation using case-based clustering and weighted association rule mining

J. Bhavithra¹ · A. Saradha²

Received: 13 December 2017 / Revised: 1 February 2018 / Accepted: 7 February 2018 / Published online: 24 February 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Recommendation system predicts and suggests those web pages that are likely to be visited by web users. The usage of recommendation system reduces delay in search and helps users to achieve the desired purpose in web search. Personalization in recommender system creates user profiles by analyzing the user's interest through previous search history and patterns. The web pages that are recommended will be predicted based on the user profile. In this paper, the idea of Case-Based Reasoning has been adapted suitable for web page recommendation as an extension of Collaborative filtering. Users' profile will be generated comprising of eight characteristic features and two content-based features generated using web access search logs. The collaboration among the k-NN user profile is identified based on Case-Based Reasoning. To enhance the accuracy Weighted Association Rule Mining is applied, which generates rules among the user profiles and optimally predicts the web pages suitable for the given search keyword by a user. To verify the effectiveness of the proposed idea, experiments were carried out with multiple datasets covering 2370 web pages accessed by 77 different users. Experiment result shows that the proposed algorithm outperforms existing methods with increased accuracy and minimum miss-out and fall-out rates.

Keywords User profile · Characteristic features · Content-based features · k-NN · Collaborative filtering · Case-based reasoning (CBR) · Weighted association rule mining (WARM)

1 Introduction

In today's web era, online users are abundant with high expectations from search engines to satisfy their query with most appropriate web pages. Current algorithms in information retrieval focus on enhancing and optimizing to achieve personalization. Based on their search on the web, query results must be customized to provide user

satisfaction. The recommendation system is one among the thriving research area today, in which personalization is done to analyze user's search interest and provide better results even for those users who do not reveal their search interest explicitly [1]. Web Mining is categorized into three different types as explained in [2]. They are Web Usage Mining, Web Structure Mining and Web Content Mining. This research is based on analysis of usage log and content in each URL logged by a corresponding user. Each web page is denoted by its corresponding URL logged. The user profile is constructed for each identical user which is determined by IP address [3]. To hide the identity of each user, their IP address is represented uniquely as random number [4]. The profile of each user was created using eight characteristic features and two content-based features, which is explained in Sect. 3.1.

Traditionally Collaborative based filtering and k-Nearest neighboring (kNN) approaches were predominantly applied in recommendation systems. Both these approaches provide recommendations for an active web user based on

✉ J. Bhavithra
bavi.rr@gmail.com
http://www.mcet.in

A. Saradha
saradha.irtt@gmail.com

¹ Department of Computer Science and Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi, Tamil Nadu 642003, India

² Department of Computer Science and Engineering, Institute of Road and Transport Technology, Erode, Tamilnadu 638316, India

other users who have similar interest and preferences [5]. Those users with similar interest are called as neighbors [6]. Unfortunately, Collaborative and kNN approaches have their own drawbacks. For example, consider a web page ‘p’ which has been recently created or modified to hold updated contents. Such page ‘p’ might not be visited/revisited by web users after it has been updated. Hence, ‘p’ may not be included for further recommendation to the currently active user. Such problem is termed as cold start problem.

The objective of this paper is to improve the accuracy of web page recommendation through the extension of kNN method to effectively recommend web page by applying Case-Based Reasoning (CBR) and Weighted Association Rule Mining (WARM) algorithms. The main focus of this paper is as follows:

- kNN algorithm is applied to identify the initial set of ‘k’ similar neighbors for any current active user. These users are referred as k-NN. Such ‘k’ neighbors are those who have the similar interest in search and target towards web pages that have similar content. For identifying ‘k’ users, collaborative filtering algorithm proposed in [7] was applied.
- CBR algorithm [8, 9, 10] is applied among ‘k’ identified neighbors, by analyzing user profiles comprised of characteristic and content-based features. CBR reduces the size of k-NN neighbors as n-NN neighbors (where $n = k/2$) resulting in reduced delay and increased performance of the system.
- To further improve the accuracy, WARM is applied following CBR. Here weights are assigned while computing support and confidence to generate more accurate rules from frequent item-sets. These final rules are used for final recommendation.

The rest of this paper is organized as follows. In Sect. 2, related work to this paper has been discussed. Section 3 discusses the concept of applying CBR for profile generation. Section 4 discusses the idea of using WARM for rule generation. Section 5 covers results and discussion. Section 6 concludes with the final findings and inferences observed in this paper.

2 Related work

Various traditional methods such as collaborative filtering, association rules, clustering, sequential patterns, hybrid methods and semantic web [11] are used for personalization and recommendation systems. Collaborative filtering developed by [7] and [12] is one of the most common approaches used for providing recommendation by finding similar users. Pearson correlation coefficient and cosine

based approach can be used to find similar users [13]. This traditional approach can still be improved by applying normal recovery Collaborative filtering [12]. But recommendation done using pure collaborative filtering approach may lead to problems such as popularity bias, cold start problem, handling dynamic pages etc. So, in order to provide personalized results, this paper combines CBR with WARM. CBR [9] generates the user profile and uses similarity knowledge to predict relevant profiles for the currently active user [9]. Such profile includes Page Rank [14] as a major feature which is computed using HITS and Page Rank algorithm. WARM is similar to traditional association rule mining and it’s more efficient as it considers the importance of transactions and item sets [15, 16].

2.1 Collaborative filtering

Collaborative filtering is one of the most common approaches used for recommendation. Collaborative Filtering systems collect visitor opinions on a set of objects using ratings, explicitly provided by the users or implicitly computed. In explicit ratings, users assign rating to items or web pages, or a positive (or negative) vote to some web pages or documents [11]. The implicit ratings are computed by considering the access to a Web page. A rating matrix is constructed where each row represents a user and each column represents an item or web page keywords [12]. Items could be any type of online information resources in an online community such as web pages, videos, music tracks, photos, academic papers, books etc. Collaborative filtering systems predict a particular user’s interest in an item using the rating matrix. Alternatively, the item–item matrix, which contains the pair-wise similarities of items, can be used as the rating matrix. Rating matrix is the basis of CF methods. The ratings collected by the system may be of both implicit and explicit forms. Although CF techniques based on implicit rating are available for recommendation, most of the CF approaches are developed for recommending items where users can provide their preferences with explicit ratings to items.

The web log files are collected from the users’ browsing history, consisting of IP address, date & time of visiting the web pages, method URL/protocol, status, received byte etc. From the log file all the web page contents are extracted, from which keywords are extracted. Page view and page rank is calculated for each URL. Based on these values, user profile is constructed. The user profile is represented in matrix format. Based on the user profile, user’s similarity is found by applying normal recovery similarity measure [12]. Collaborative filtering approach called Normal Recovery Collaborative Filtering (NRCF) is applied on similar users obtained, for web page recommendation. When new user enters a search query same as other similar

user query, then the webpages visited by similar users are recommended to the new user. Normal recovery similarity measure is applied on the users profile and more similar users to calculate the degree of similarity between two users using the following Eq. (1) stated in [12, 17]:

$$\text{Sim}(u, v) = 1 - \frac{\sqrt{\sum_{i \in I} \left(\frac{r_{u,i} - r_{u,\min}}{r_{u,\max} - r_{u,\min}} - \frac{r_{v,i} - r_{v,\min}}{r_{v,\max} - r_{v,\min}} \right)^2}}{\sqrt{|I|}} \quad (1)$$

where, i is the set of web pages that are co-visited by user u and v . $|I|$ is the number of i , i.e. total number web pages co-visited by users u and v . $r_{u,i}$ is the value of web page keyword and time spent in particular web page from user u in user web page matrix. $r_{u,\min}$ and $r_{u,\max}$ are the lowest and highest values of user u . $r_{v,\min}$ and $r_{v,\max}$ are the lowest and highest values of user v [12].

2.2 Content-based recommendation

Content-based filtering is a type of information extraction system, where web pages are extracted based on the semantic similarity between the content in those web pages visited by users in past history [18, 19]. Web content mining applications mostly rely on content-based filtering approaches. Content-based filtering offers predominant support for web page recommendation system. In this technique, the keywords and its frequency of occurrence in those web pages that were previously visited are collected. Then, the semantic similarity between such keywords will be analyzed for further process. For example, consider two users “u1” and “u2” who frequently visit web pages based on their domain of interest. Let u1 always focus on health related web pages and u2 focus on gadget-related sites. Now, during the real time if any active academic user “ua” search for the query “apple”, he will be mostly related to apple devices based sites, rather than apple fruit. So, he will be recommended the sites referred by u2. Similarly, when a dietician “ub” searched for “apple” he will be recommended the sites referred by u1.

Recommendation engine classifies “ua” as an academic user and “ub” as a dietician based on the contents (keywords) of the web pages navigated in past history. Along with the keywords, the semantic similarity between them is also analysed for more effective domain grouping. Content-based classification is used for grouping web users under various domains. For such classification, the frequency and keywords in web pages are represented using TF/IDF notations. TF corresponds to Term Frequency and IDF corresponds to Inverse Document Frequency. The following Eqs. (2), (3) and (4) are used to determine TF-IDF [20] of a term j within a document collection N .

$$\text{tf-idf}(j) = \text{tf}(j) \times \text{idf}(j) \quad (2)$$

where,

$$\text{tf}(j) = \text{Frequency of term } j \text{ in a document} \quad (3)$$

and,

$$\text{idf}(j) = \log \left(\frac{N}{\text{No. of docs that has } j \text{ at least once}} \right) \quad (4)$$

3 Case based clustering for web page recommendation

3.1 Feature selection

Case-based clustering applies case based reasoning (CBR) for clustering the user profiles. CBR is a process of finding solutions to new problems based on the solutions of similar past problems [8, 10]. In this paper, the phenomenon of such CBR is intended to be applied in Web page recommendation system [21]. Here, a user profile that narrates user interest, searching pattern and web accessing phenomena are created. Such user profile comprises of the following ten features [22, 21].

- Time on page (TOP)
- Time on site (TOS)
- Average time at this page (ATP)
- Bounce rate (BR)
- Exit rate (ER)
- Conversion rate (CR)
- Number of visitors (NOV)
- Average page rank (APR)
- Top similar keywords (SK)
- Average similarity between keywords (ASM)

In the paper [22] eight characteristic features along with a content-based feature called top similar keywords (SK) were defined. The proposed algorithm introduces another content-based feature called “Average Similarity between keywords” in order to increase the accuracy of recommendation. Hence, the proposed system uses ten characteristic and content-based features for the development of user profile. The methodology for the identification of such features from user’s web access log file is described in Sect. 3.2.

Another contribution of the proposed algorithm is introducing the concept of employing weights (β) for each feature while developing the user profile. The advantage of adding weight is to give more strength to selective features that help in enhancing the accuracy of predicting web pages for recommendation. In the proposed system, the value of β ranges between 1.0 and 2.0. The idea here is to

double ($\beta = 2$) the contribution of most significant features, considerably increase ($\beta = 1.75$) the strength of significant features, marginally increase ($\beta = 1.5$) the weight of most relevant features and maintain ($\beta = 1.0$) the contribution of required features in a user profile to enhance the accuracy of prediction. The following Table 1 shows the weight (β) assignment of all features for developing the user profile. Initially, traditional collaborative filtering approach is used to filter “k” number of users (neighbors) from the global set of web users. The value “k” is a level of threshold which can be set by recommendation engine to balance between optimization and increasing search accuracy.

Now, the profile of all “k” users are analyzed and compared with current active user’s profile as narrated in Table 2. Analysis of user profiles has been done based on CBR Approach [21]. Here, the selected ten features of “k” users are compared by calculating the similarity with current Active User (AU). From all such similar users, top N users profiles whose similarity is below the threshold are

selected. In the proposed system, the threshold value is set dynamically as the following Eq. (5):

$$\text{Threshold} = \frac{1}{k} \sum_{i=1}^k \text{Sim}(\text{AU}, U_i) \tag{5}$$

WARM algorithm is then applied to generate rules that filter the list of all web pages (URLs) that were visited mostly by N users filtered by CBR approach. The working principle of CBR in web page recommendation with an example of $k = 4$ users is shown in the following Table 2 which is generated based on sample training dataset 1 (discussed under Sect. 6).

For experimentation, AOL web access log dataset [23] was used. The log file contains web query log data from ~ 650 k users. In order to have privacy preservation, IP addresses of individual users are represented using anonymous IDs. Hence each user is represented by unique ID. The schema of this log dataset is: {AnonID, Query, Query Time, Item Rank, ClickURL}. Where, AnonID

Table 1 Assignment of weights (β) for each feature

S. No	Features in user profile	β value (1.0–2.0)
1.	Time on page (TOP)	1.00
2.	Time on site (TOS)	1.00
3.	Average time at this page (ATP)	1.75
4.	Bounce rate (BR)	1.75
5.	Exit rate (ER)	1.50
6.	Conversion rate (CR)	1.75
7.	Number of visitors (NOV)	1.50
8.	Average page rank (APR)	2.00
9.	Top similar keywords (SK)	1.75
10.	Average similarity between keywords (ASM)	2.00

Table 2 Working principle of CBR based clustering approach (where $k = 4$)

Features in user profile	Assigned weights	Similar existing User 1	Similar existing User 2	Similar existing User 3	Similar existing User 4	Active User (AU)
UID	NA	841	7895	87	785	7999
Time on Page (TOP) (s)	1.75	140	126	195	183	169
Time on Site (TOS) (s)	1.00	158	139	187	176	153
Average time at this page (ATP) (s)	1.00	58	12	18	43	37
Bounce rate (BR) (%)	1.75	58%	49%	68%	61%	54%
Exit rate (ER) (%)	1.50	38%	23%	53%	41%	42%
Conversion rate (CR) (%)	1.75	4.17%	2.56%	5.28%	3.94%	4.23%
Number of visitors (NOV) (Nos)	1.50	14	6	3	9	7
Average page rank (APR) (Nos)	2.00	6	1	3	4	8
Top similar keywords (SK) (Nos)	1.75	254	69	124	176	185
Average similarity between keywords (ASM) (Nos)	2.00	158	69	85	248	173
Similarity with AU		3814	15,000	7925	3546	NA

represents an anonymous user ID number to preserve user privacy. Query denotes the query issued by the user. Query Time says the time at which the query was submitted for search. Item Rank denotes that if the user clicked on a search result, the rank of the item on which they clicked is listed. Finally, Click URL represents the domain portion of the URL that the user clicked on a search result. In the pre-processing stage, the log file is cleansed by removing unwanted information such as blocked URLs, inappropriate and incomplete entries. Finally, the user profile is constructed by analyzing the search pattern and URLs of each individual user identified using AnonID.

3.2 User profile generation

The user profile based on eight characteristic features and two content-based features are created as explained below:

3.2.1 Time on page (TOP)

The parameter Time on Page is the total time spent by an active user within a particular page. An average of time spent on all web pages is measured using the following algorithm 1.

Algorithm 1: Computing Time on Page (τ_{Avg})

```

-----
Begin
Input:  $DS_U$  – pre-processed log dataset of each user
       $Max_T$  – Maximum Threshold
       $Min_T$  – Minimum Threshold
Output: A hash table  $\langle URL, \tau_{pi} \rangle, \tau_{Avg}$ 
Begin
For each user dataset  $DS_U$  do the following
For each URL  $i$  to  $n$  in  $DS_U$ 
start_qt  $\leftarrow$  Query Time( $URL_i$ )
end_qt  $\leftarrow$  Query Time( $URL_{i+1}$ )
diff_qt  $\leftarrow$  end_qt – start_qt
if (diff_qt <  $Max_T$ ) and (diff_qt >  $Min_T$ )
 $\tau_{pi} \leftarrow$  diff_qt
sum = sum+  $\tau_{pi}$ 
      Update hash table  $\langle URL_i, \tau_{pi} \rangle$ 
End For
 $\tau_{Avg} = sum/n$ 
End
-----

```

3.2.2 Time on site (TOS)

The time spent by individual user within a website is computed as TOS. This time is calculated by the following Eq. (6) and Eq. (7); where, the time spent on each page pi with same hostnames (URIs) are summed together to identify τ_S .

$$\tau_S (URI) = \sum_{uri \in \langle URL, \tau_{pi} \rangle} \tau_{pi} (URI) \quad (6)$$

$$\tau_{SAvg} = \frac{1}{n} \sum_{i=1}^n \tau_S (URI_i) \quad (7)$$

3.2.3 Average time on this page (ATP)

The average time spent by the corresponding user for any page pi is identified using Eq. (8); Where pi is the page for which average time spent is to be calculated. URL is the entire list of web page URLs visited by web users, τ_{pi} is time spent for each page pi and N is the total number of occurrences of the page pi .

$$\overline{x(pi)} = \frac{\sum_{pi \in URL} \tau_{pi}}{N} \quad (8)$$

3.2.4 Bounce rate (BR)

The web page access percentage with respect to session wise grouping of access pattern is called as bounce rate. Today BR plays a vital role in web analytics. Web pages access pattern is grouped into sessions based on date and time difference between two consecutive page requests. If the date and time difference is exceeding certain time limit of 10 min, the access patterns were grouped as clusters called as sessions. The page pi 's access rate between all such sessions is computed as BR using the Eq. (9); Where 's' represents each session from the complete set of sessions 'S' and TS represents the total number of sessions active by a web user. NS(pi) denotes the total number of sessions where page pi has been accessed.

$$BR(pi) = \frac{\sum_{s \in S} \sum_{(pi \in s) \cap (i=1)} \tau_{pi}}{NS(pi)} \times TS \quad (9)$$

3.2.5 Exit rate (ER)

The rate at which, the web page (pi) will be at the end of the session is computed as ER. Here, the occurrence of pi being the last entry within the session is calculated to identify the exit rate using the Eq. (10)

$$ER(pi) = \frac{\sum_{s \in S} \sum_{(pi \in s) \cap (i=N)} \tau_{pi}}{NS(pi)} \times TS \quad (10)$$

3.2.6 Conversion rate (CR)

The conversion rate for each web page is computed as the ratio between total sessions accessed by a user to the total number of sessions that contains the page pi . Equation (11)

computes the conversion rate of page p_i . Here, TS denotes the total number of sessions grouped under each user. $NS(p_i)$ denotes the number of sessions contains page p_i .

$$CR(p_i) = \frac{TS}{NS(p_i)} \times 100 \tag{11}$$

3.2.7 Number of visitors (NV)

The total number of visitors, also called as page views, for each web page visited by the corresponding user has to be computed to analyze the priority of a web page. If more number of users has been visiting, the corresponding page is given with good preference for further recommendation. The number of visitors for a particular page p_i is computed using Eq. (12):

$$NV(p_i) = \sum_{j=1}^N \sum_{p_i \in S_j} n \tag{12}$$

where,

$$n = \begin{cases} 0, & \text{if } p_i \text{ is not present atleast once in } S_j \\ 1, & \text{if } p_i \text{ is present atleast once in } S_j \end{cases}$$

3.2.8 Total page rank (TPR)

Page rank is a numerical value that measure’s a webpage importance among the group of similar web pages. Such page rank is computed based on Random Surfer model [14]. This algorithm computes the page rank based on link structure of the web page [24, 25]. A page gets hold of high rank if the addition of the ranks of its backlinks is high. The rank of the given page is thus computed using the following Eq. (13)

$$TPR = \frac{1}{N} \left[(1 - d) + d \sum_{v \in B(u)} Page_{wt} \times \frac{PR(v)}{N_v} \right] \tag{13}$$

Where, u represents a web page. $B(u)$ is the set of pages that point to u . $PR(v)$ is the page rank of page v that points to page u . N_v is the number of outgoing links of page and d is the damping factor that is set between 0 and 1. The damping factor is the decay factor that represents the chance of a user stop clicking links within a current page and then requesting another random page [14]. $Page_{wt}$ is termed as Page weight which is calculated based on frequency and duration as in Eq. (14).

$$Page_{wt}(PW) = NV(p_i) \times \tau p_i \tag{14}$$

where τp_i is total time spent by the user on particular webpage represented by Algorithm 1. A quick jump might also occur due to the short length of a web page so the size

of page may affect the actual visiting time. Hence, duration is normalized by the length of the web page, i.e. the total bytes of the page. $NV(p_i)$ is the number of times that a page is accessed by different users; computed by Eq. (12).

3.2.9 Top similar keywords (SK)

The top similar keywords under each ranked page p_i are considered for further recommendation. To identify such top keywords, tokenization and stemming process are performed. The following algorithm (2) is used to identify top keywords based on their frequency of occurrence.

Algorithm 2: Finding Top Similar Keywords (SK)

```

-----
Begin
Input: DSU – pre-processed log dataset of each user
      K – number of top keywords required
Output: A hash table <Keyword, frequency>
Begin
  Initialize HT as a tuple <keyword, frequency>
  For each web page  $p_i$  in DSU do the following
    Read the contents of  $p_i$ 
    tok[] ← Tokenization of contents
    Stem_tok[] ← Stemming(tok[])
    Tokens[] ← Remove StopWords(Stem_tok[])
    For each token tok in Tokens[]
      i ← number of times tok occurs
      if tok ∈ HT
        search tok in HT<keyword>
        Update frequency ← frequency+1
      else
        Add <tok,1> in HT
    End for
  Open HT<key, freq>
  Arrange HT in descending order of freq
  Return HT<Key,freq>
  Return Top “K” keywords from HT<Key,freq>
End
-----

```

3.2.10 Average similarity between keywords (ASM)

The set of top keywords gathered using Algorithm 2 for all “k” users are further investigated to find the semantic similarity between each user and current Active User (AU). This similarity is used to find the distance between two users based on their search interest. The following algorithm (3) is used to find the average similarity between keywords.

Algorithm 3: Average Similarity between Keywords (ASM)

```

Begin
Input: HT < Keyword, frequency> of AU
HTU < Keyword, frequency>[] of “k” users

Output: An array Ave_Sim[] for “k” users
Begin
KeyTable[][] ← Merge(Keywords, Freq, kusers)
For each user from i=1 to k
  For each keyword from j=1 to m
    w[i..m] ← HT<Keyword>
    f[i..m] ← HT<frequency>
// find the similarity between AU and neighbor

Sim[i][j] =  $\sqrt{\sum_{j=1}^m (f(j) \times (\text{KeyW}[i][j] - w[j]))}$ 

  End For
End For
// Finding Average for top k similar keywords
k_sim[][] ← Arrange in decreasing order of similarity value
for each user from i=1 to k

  Ave_sim[i] =  $\frac{1}{m} \sum_{j=1}^m k\_sim[i][j]$ 

End For
Return Ave_sim[] of “k” users.
End

```

3.3 Finding similarity score

Finally, the similarity among “k” existing (EU) user profiles and the current Active User (AU) is to be identified to filter the most similar neighbors. To identify this similarity, the following Eq. (15) is used.

$$\text{sim}(AU, EU) = \sum_{f=1}^{10} \left[\beta_f \times (EU_f - AU_f)^2 \right] \quad (15)$$

where, f denotes the ten features retrieved from individual user profiles. The similarity score is determined by the Euclidean distance between each existing user ($EU_{1..k}$) and current active user (AU). This difference is multiplied by the weights (β) assigned as in Table 1. After calculating the similarity scores, the threshold value is determined as stated in Eq. (5). Finally, the most similar users whose similarity value is lesser than the threshold value will be selected for further analysis using WARM algorithm which is discussed in the following section. Thus CBR has been applied in order to reduce the k -nearest neighbors and thereby selecting the most similar n -nearest neighbor users. As k -NN has been reduced to n -NN the proposed CBR based recommendation system was found to be working with enhanced performance and speed.

4 Recommendation using WARM

4.1 Identifying frequent item set

To further enhance the accuracy of recommendation, Weighted Association Rules Mining algorithm is used. Following the CBR process, Association rules are mined considering the n -NN neighbor users. Association Rule mining is another predominant algorithm used for effective product recommendation [24, 25]. Here, weights have been computed using Eq. (20) for each item (web page) that will be analyzed for recommendation to any active user. Hence the rule mining algorithm is termed as Weighted Association Rule Mining. Here, n -NN user’s most visited pages (fetched from their profile) that were matching the current user’s query (Eq. (16)) are mined to find frequent item-set which is called as set S . Association rules are generated based on the frequent item-set [26]. The set of web pages that contains the query word(s) is filtered and called as S' represented using Eq. (17).

$$S = \{p_1, p_2, p_3, \dots, p_s\} \quad (16)$$

$$S' = \{p_i, p_j, p_k, \dots, p_n\} \quad (17)$$

$$\{p_i, p_j, p_k\} \Rightarrow \{p_m\} \quad (18)$$

For example, consider the following Eq. (18). The above rule states that users those who visited web pages “ p_i ”, “ p_j ” and “ p_k ”, in any order, they are most likely to visit web page “ p_m ”. Hence it might be most appropriate to recommend web page “ p_m ” to the currently active user. Here pages p_i , p_j , p_k and p_m are termed as frequent item-sets. Association rules of type mentioned in Eq. (18) are mined using those frequent item-sets from set S . Support and Confidence value for all frequent item-sets “ x ” that constitutes to those association rules mined are computed to eliminate rules that are not suitable for recommendation process [27]. The support and confidence value of each mined rule is computed using the following Eq. (19) and Eq. (21):

$$\text{Support}(x) = \text{Wt}(x) \times \frac{|\{S' \in S; x \subseteq S'\}|}{|S|} \quad (19)$$

where, $\text{Wt}(x)$ is the weight of all web pages contained in the item-set x . $\text{Wt}(x)$ is computed as Eq. (20):

$$\text{Wt}(x) = \sum_{p_i \in x} \frac{\text{BR}(p_i) + \text{ER}(p_i) + \text{CR}(p_i)}{300} \quad (20)$$

where, $\text{BR}(p_i)$, $\text{ER}(p_i)$ and $\text{CR}(p_i)$ were the Bounce Rate, Exit rate and Conversion rates of webpage p_i .

The confidence of any rule $p_1 \Rightarrow p_2$ will be computed using the following Eq. (21):

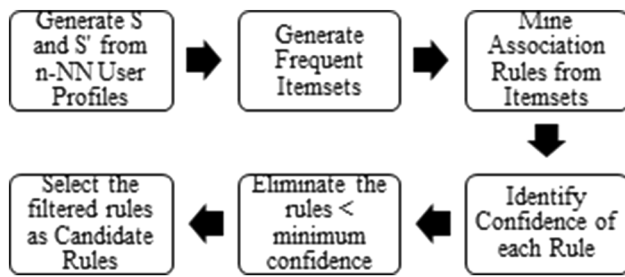


Fig. 1 Process of generating weighted rules from user profiles

$$\text{Confidence } (p1 \Rightarrow p2) = \frac{\text{Support}(p1 \cup p2)}{\text{Support } (p1)} \tag{21}$$

4.2 Generation of association rules

The overall rule generation process is described using the following Fig. 1. The association rules that are generated will be sorted based on decreasing confidence value, enabling the most appropriate rules with high confidence may be listed at the top. The candidate rules that were mined with maximum confidence value for each query keyword given by end user is listed in Table 3.

4.3 Recommendation process

Finally, the top “m” rules that are ranked based on confidence value are selected for recommendation process. Here “m” is set by web server/search-engine. Various experiments were conducted with varying m values as 30, 40 and 50 to analyse the accuracy of the proposed algorithm. The RHS from each “m” rules are selected as final web pages to be recommended for the end user.

5 Results and discussion

5.1 Data set

For the research experimentation and analysis, AOL log dataset has been used. The log file contains web query log data from ~ 650 k users. In order to have privacy

preservation, IP addresses of individual users are represented using anonymous ID. Hence each user is represented by unique ID. The experiments were carried out with datasets covering 7175 web pages accessed by 287 different users. The schema of this log dataset is: {AnonID, Query, Query Time, Item Rank, ClickURL} [23]. Where, Where, AnonID represents an anonymous user ID number to preserve user privacy [28, 29]. Query denotes the query issued by the user. Query Time says the time at which the query was submitted for search. Item Rank denotes that if the user clicked on a search result, the rank of the item on which they clicked is listed [30, 31]. Finally, Click URL represents the domain portion of the URL that the user clicked on a search result. The web access log dataset is divided into seven samples of equal size with 50 records as mentioned in the following Table 4:

5.2 Evaluation metrics

In order to verify the performance of the proposed algorithm, the following metrics were identified: F1-Measure, Miss-Rate (MR), Fallout Rate (FR) and Matthews Correlation [32]. In order to compute these evaluation metrics, the following Table 5 is developed.

5.2.1 F1-measure

The F1-Measure is computed based on two metrics such as Precision or True Positive Accuracy (Confidence) and Recall or True Positive rate [32]. The Precision is defined as the ratio between the recommended web pages that are relevant to the user query to the total number of recommended items [33–35]. Precision is represented using Eq. (22).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{22}$$

Recall is calculated as per Eq. (23) and is defined as the ratio of web pages recommended that are relevant to the total number of relevant webpages [32] considered for experimentation purposes

$$\text{Recall} = \frac{TP}{TP + FN} \tag{23}$$

Table 3 Candidate Rules (with maximum confidence value) for the given query

Query keyword	Generated rules
Books	{pearsoned.com, pearsonhighered.com} ⇒ {pearsonassessments.com}
Shoes	{amazon.com, snapdeal.com, ebay.com} ⇒ {flipcart.com}
Online courses	{coursera.com, openuniversity.edu, Alison.com} ⇒ {edx.org}
food recipes	{food.ndtv.com, bbcgoodfood.com, countryliving.com} ⇒ {indianfoodforever.com}
News	{thehindu.com, indiatoday.in} ⇒ {timesofindia.com}
Live score	{espnricinfo.com, cricbuzz.com} ⇒ {livescore.in}

Table 4 Various sample datasets used for experimentation

Sample category	Description
Sample 1	Without any conditions, access log of 50 users were selected randomly
Sample 2	Uniform sampling was performed to select one user after each 50 records
Sample 3	The query was analyzed and categorized into various domains. 50 users accessed under academic category were selected
Sample 4	The top 50 users who access web frequently were selected based on the maximum length (no. of URLs) within each session
Sample 5	The top 50 users who do not access web frequently were selected based on the minimum length (no. of URLs) within each session
Sample 6	The top 50 users having profile with maximum number of identical search keywords were selected
Sample 7	The top 50 users having profile with minimum number of identical search keywords were selected

Table 5 Contingency table used to compute precision and recall

Category	Remarks
True positive (TP)	The web pages that are recommended were relevant
False positive (FP)	The web pages that are recommended were irrelevant
True negative (TN)	The web pages that are not recommended were irrelevant
False negative (FN)	The web pages that are not recommended are relevant

The specifications of TP, FP, TN and FN are stated in Table 1 [32]. These Precision and recall values are used to compute F1-measure as given in Eq. (24).

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (24)$$

5.2.2 Miss rate (MR)

The miss rate is calculated based on the total number of relevant web pages that were not recommended [32]. This is also termed as False Negative Rate as denoted in Eq. (25).

$$MR = \frac{FN}{TP + FN} \quad (25)$$

5.2.3 Fall out rate (FR)

The false positive rate (calculated using Eq. (26)) or Fall-out Rate is defined as the rate of irrelevant pages that were recommended to the total number of irrelevant pages [32].

$$FR = \frac{FP}{FP + TN} \quad (26)$$

5.2.4 Matthews correlation (MC)

The Matthews Correlation is used to analyze the effectiveness of the proposed classification algorithm [32]. This is computed using the Eq. (27).

$$MC = \frac{(TP \cdot TN) - (FP \cdot FN)}{\sqrt{(TP + FN) \cdot (FP + TN) \cdot (TP + FP) \cdot (FN + TN)}} \quad (27)$$

5.3 Experiment results

Experiments were conducted using the seven samples of dataset running under three algorithms: Collaborative Filtering (CF); Case-Based Reasoning (CBR) and Case-Based Reasoning with Weighted Association Rule Mining (CBR with WARM). The graphs that measure F1-Measure, Miss Rate (MR), Fallout Rate (FR) and Matthews Correlation (MC) were shown in Figs. 2, 3, 4 and 5 respectively.

The F1-measure has been analyzed with varying values of “k” as k = 5; k = 10; k = 15; k = 20; k = 25; k = 30; k = 35 and k = 40. Figure 2 clearly states that in all the algorithms with various samples of dataset tested, the optimum value for “k” lies within 20–25 with increased F1-measure. In order to verify the error possibility of the proposed CBR with WARM algorithm, Miss Rate and Fallout rate were tested by conducting experiments with the same seven sample datasets as described in Table 4.

Figures 3 and 4 shows that the Miss Rate or False Negative rate and the Fallout Rate or False Positive rate for the proposed algorithm has been reduced when compared to CBR system and the existing traditional collaborative filtering approaches. Figure 5 analyses the effectiveness of the proposed CBR with WARM algorithm using Matthews

Fig. 2 Comparing F1-measure with varying “k” values (from k = 5 to 40)

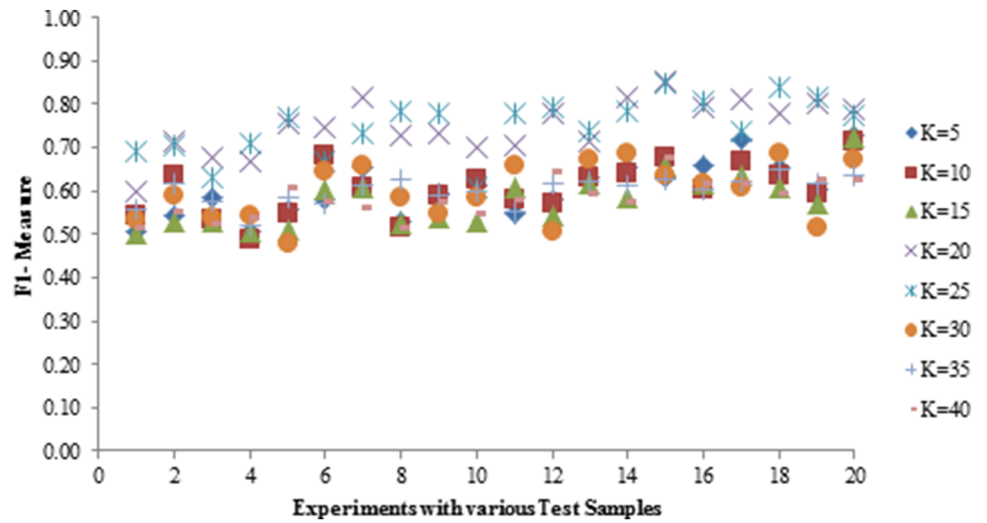


Fig. 3 Testing miss rate of three algorithms with seven data samples

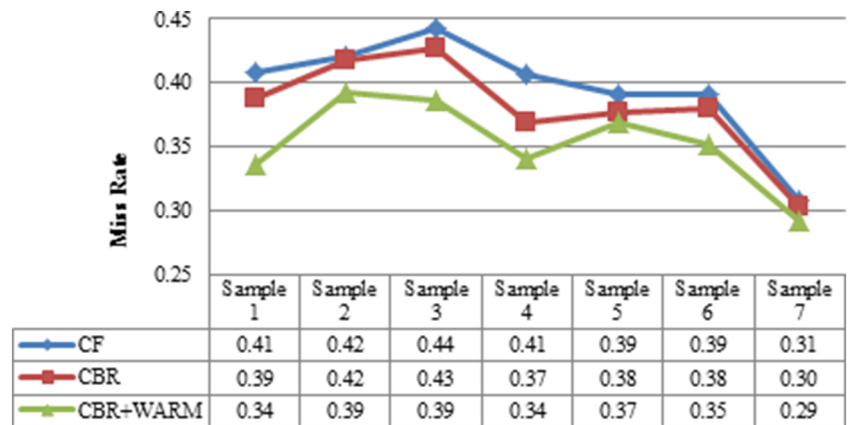


Fig. 4 Testing fallout rate of three algorithms with seven data samples

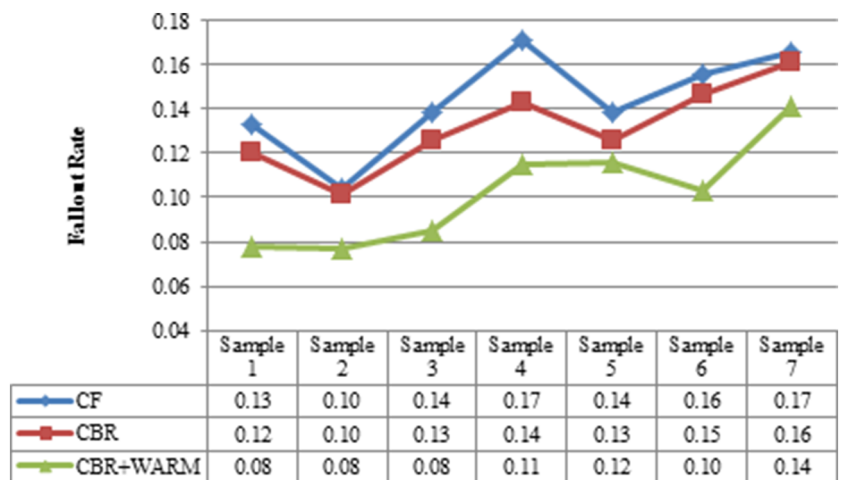
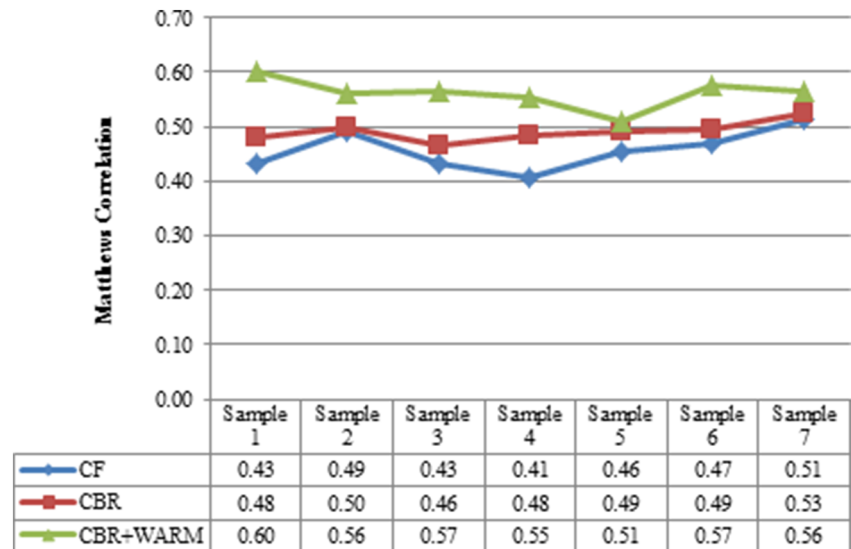


Fig. 5 Comparing the efficiency of various algorithms using Matthews Correlation



Correlation (MC). It was found that, the proposed algorithm outperforms the other two approaches.

6 Conclusions

In this paper, a novel approach to develop user profiles was proposed where eight characteristic features and two content-based features were identified for efficient classification of user profiles. In addition, a new algorithm based on Case-Based Reasoning was proposed that enhances the performance of Collaborative filtering based web page recommendation system. In order to further optimize and increase the accuracy of recommendation process, Weighted Association Rule Mining approach was applied along with CBR. To analyze the effectiveness of proposed algorithms, experiments were conducted on seven test case samples for three algorithms namely Collaborative Filtering, Case-Based Reasoning, Case-Based Reasoning with Weighted Association Rule Mining Algorithm. The experiment result with AOL dataset concludes that optimal 'k' value for selecting neighbors for CBR approach lies within 20 to 25. The error level of proposed algorithm was found to have minimum Miss Rate and Fallout Rate. In terms of classification efficiency, the proposed system was found to outperform existing method.

References

- Ahmad, A.M., Hijazi, M.H.A.: Web Page Recommendation Model for Web Personalization, pp. 587–593. Springer, Berlin (2004)
- Chu, W., Lin, T.Y.: Foundations and Advances in Data Mining—Studies in Fuzziness and Soft Computing, vol. 180. Springer, Berlin (2005)
- Shrivastava, A., Shukla, N.: Extracting knowledge from user access logs. *Int. J. Sci. Res. Publ.* **2**(4), 1 (2012)
- Al Murtadha, Y., Sulaiman, N.B., Mustapha, Y.N., Udzir, N.I., Muda, Z.: Web page recommendation system for anonymous users based on web usage mining. *Adv. Commun. Comput. Syst. Circ. Devices*. ISBN: 978-960-474-250-9 (2011)
- Goksef, M., Gunduz-Oguducu, Ş.: Combination of web page recommender systems. *Elsevier J. Exp. Syst. Appl.* **34**, 2911–2922 (2010)
- Zuva, T., Ojo, S.O., Ngwira, S.M., Zuva, K.: A survey of recommender systems techniques: challenges and evaluation metrics. *Int. J. Emerg. Technol. Adv. Eng.* **2**(11), 382–386 (2012)
- Zheng, Z., Ma, H., Lyu, M.R., King, I.: Wsrec: a collaborative filtering based web service recommender system. In: *IEEE International Conference on Web Services*, pp. 437–444 (2009)
- Patel, D.K.: A retrieval strategy for case-based reasoning using USIMSCAR for hierarchical case. *Int. J. Adv. Eng. Res. Technol.* **2**(2), 65–69 (2014)
- Kang, Y.-B., Krishnaswamy, S., Zaslavsky, A.: A retrieval strategy for case-based reasoning using similarity and association knowledge. *IEEE Trans. Cybern.* **44**, 473–487 (2014)
- Tyagi, S., Bharadwaj, K.K.: A hybrid recommender system using rule-based and case-based reasoning. *Int. J. Inf. Electron. Eng.* **2**(4), 586–590 (2012)
- Gunduz-Oguducu, Ş.: Web page recommendation models: theory and algorithms. *Synth. Lect. Data Manag.* **2**, 1–85 (2010)
- Sun, H., Zheng, Z., Chen, J., Lyu, M.R.: Personalized web service recommendation via normal recovery collaborative filtering. *IEEE Trans. Serv. Comput.* **6**, 573–579 (2012)
- Zheng, Z., Ma, H., Lyu, M.R., Kin, I.: QoS-aware web service recommendation by collaborative filtering. *IEEE Trans. Serv. Comput.* **4**, 140–152 (2011)
- Devi, P., Gupta, A., Dixit, A.: Comparative study of HITS and PageRank link based ranking algorithms. *Int. J. Adv. Res. Comput. Commun. Eng.* **3**, 5749–5754 (2014)
- Shashichhikara, Sharma, P.: Weighted association rule mining: a survey. *Int. J. Res. Appl. Sci. Eng. Technol.* **2**, 84–88 (2014)
- Yan, L., Li, C.: Incorporating page view weight into an association-rule-based web recommendation system. In: *Proceedings of*

- 19th Australian Conference on Advances in Artificial Intelligence, pp. 577–586 (2006)
17. Sun, H., Peng, Y., Chen, J., Liu, C., Sun, Y.: A new similarity measure based on adjusted Euclidean distance for memory-based collaborative filtering. *J. Softw.* **6**, 993–1000 (2011)
 18. Niki, R., Kapadia, Patel, K.: Web content mining techniques—a comprehensive survey. *Int. J. Res. Eng. Appl. Sci.* **2**(2): 1869, 1877 (2012)
 19. Yao, L., Sheng, Q.Z., Ngu, A.H.H., Yu, J., Segev, A.: Unified collaborative and content-based web service recommendation. *IEEE Trans. Serv. Comput.* **8**(3), 453–466 (2015)
 20. Gunduz-Oguducu, Ş., Tamer Ozsu, M.: Incremental click-stream tree model: learning from new users for web page prediction. *Distrib. Parallel Databases* **19**, 5–27 (2006)
 21. Bhavithra, J., Saradha, A., Aiswarya, J.: An improved mechanism for user profiling and recommendation using case-based reasoning. *IIOAB J.* **8**(2), 319–327 (2017)
 22. Abirami, S., Bhavithra, J., Saradha, A.: A web page recommendation using Naïve Bayes algorithm in hybrid approach. In: Proceedings of the IEEE International Conference on Science, Technology, Engineering and Management (ICSTEM), March 2017
 23. Noll, M.G.: AOL Research Publishes 650,000 User Queries, 2006. <http://www.michael-noll.com/blog/2006/08/07/aol-research-publishes-500k-user-queries/>
 24. Forsati, R., Meybodi, M.R., Ghari Neiat, A.: Web page personalization based on weighted association rules. In: IEEE International Conference on Electronic Computer Technology, Macau, China, pp. 130–135 (2009)
 25. Wanaskar, U.H., Vij, S.R., Mukhopadhyay, D.: A hybrid web recommendation system based on the improved association rule mining algorithm. *J. Softw. Eng. Appl.* **6**, 396–404 (2013)
 26. Jiang, T., Tan, A.H., Wang, K.: Mining generalized associations of semantic relations from textual web content. *IEEE Trans. Knowl. Data Eng.* **19**(2), 164–179 (2007)
 27. Chen, Y., Wu, C., Xie, M., Guo, X.: Solving the sparsity problem in recommender systems using association retrieval. *J. Comput.* **6**(9), 1896–1902 (2011)
 28. Pierrakos, D., Paliouras, G.: Personalizing web directories with the AIDOF web usage data. *IEEE Trans. Knowl. Data Eng.* **22**(9), 1331–1344 (2010)
 29. Nguyen, T.T.S., Lu, H.Y., Lu, J.: Web-page recommendation based on web usage and domain knowledge. *IEEE Trans. Knowl. Data Eng.* **26**(10), 2574–2587 (2014)
 30. Pu, P., Chen, L., Hu, R.: Evaluating recommender systems from the user's perspective: survey of the state of the art. *User Model. User-Adap. Inter.* **22**(4–5), 317–355 (2012)
 31. Al Murtadha, Y., Sulaiman, N.B., Mustapha, N., Udzir, N.: Improved web page recommendation system using profile aggregation based on clustering of transactions. *Am. J. Appl. Sci.* **8**(3), 277–283 (2010)
 32. Schröder, G., Thiele, M., Lehner, W.: Setting goals and choosing metrics for recommender system evaluations. In: Proceedings of the Second Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI 2) (2011)
 33. Lu, Z., Zha, H., Yang, X., Lin, W., Zheng, Z.: A new algorithm for inferring user search goals with feedback sessions. *IEEE Trans. Knowl. Data Eng.* **25**(3), 502–513 (2013)
 34. Eirinaki, M., Lampos, C., Paulakis, S., Vazirgiannis, M.: Web Personalization Integrating Content Semantics and Navigational Patterns, WIDM'04, ACM (2004)
 35. Anand, S.S., Mobasher, B.: Intelligent Techniques for Web Personalization, pp. 1–36. Springer, Berlin (2005)



J. Bhavithra completed her Bachelor of Engineering during 2003 and Master of Engineering during 2005, specialized in Computer Science and Engineering. She is currently pursuing research under Information and Communication Engineering specialized in Web Page Recommendation systems and Machine Learning Techniques. She is currently working as Assistant Professor (Senior Scale) in the department of Computer Science and Engineering at Dr. Mahalingam College of Engineering and Technology, Pollachi, Coimbatore (DT), Tamil Nadu, India.



A. Saradha completed her doctorate in Information and Communication Engineering specialized in the field of Pattern Recognition. She is currently working as Professor and Head in the Department of Computer Science and Engineering, Institute of Road and Transport Technology, Erode, Tamil Nadu, India. She has more than 25 years of teaching and research experience. Her research interests are Web Mining, Human Computer Interaction and Pattern Recognition.