Decision Support Systems 75 (2015) 1–10

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

A web recommendation system considering sequential information

Rajhans Mishra [a,*], Pradeep Kumar [b], Bharat Bhasker [b]

[a] Indian Institute of Management, Indore, Information Systems Department, Indore, India
[b] Indian Institute of Management, Lucknow, IT and Systems Department, Lucknow, India

ARTICLE INFO

Article history:
Received 23 March 2013
Received in revised form 29 January 2015
Accepted 6 April 2015
Available online 14 April 2015

Keywords:
Web recommendation system
Soft clusters
Sequential information
Singular value decomposition (SVD)

ABSTRACT

With the rapid growth of information technology, the current era is witnessing an exponential increase in the generation and collection of web data. Projecting the right information to the right person is becoming more difficult day by day, which in turn adds complexity to the decision making process. *Recommendation systems* are intelligent systems that address this issue. They are widely used in e-commerce websites to recommend products to users. Most of the popular recommendation systems consider only the content information of users and ignore sequential information. Sequential information also provides useful insights about the behavior of users. We have developed a novel system that considers sequential information present in web navigation patterns, along with content information. We also consider soft clusters during clustering, which helps in capturing the multiple interests of users. The proposed system has utilized similarity upper approximation and singular value decomposition (SVD) for the generation of recommendations for users. We tested our approach on three datasets, the *MSNBC* benchmark dataset, simulated dataset and CTI dataset. We compared our approach with the first order Markov model as well as random prediction model. The results validate the viability of our approach.

1. Introduction

Massive digitization of data and affordable processing capabilities have motivated organizations to shift from the traditional world of mass production to the new world of customization with respect to their offerings [49]. The development of e-commerce platforms has allowed companies to provide more options to the customers. Decision support systems are required to collect the huge amount of data, process it and project it to the managers in support of customization. Recommendation system is a decision support system which can provide the desirable information to the customers as per their needs.

Recommendation systems are used by e-commerce organizations to suggest products to their customers. The products can be recommended, based on the top sellers on a site, demographics of the customer, analysis of the past buying behavior of the customer, etc. Recommendation systems generate recommendations for the users by exploring their requirements and likes. They generate different recommendations to suit different users, thus providing customized web interface to the users. Thus, the web is personalized for each user using a recommendation system. Currently most of the e-commerce organizations have enabled recommendation systems at the back end, offering web recommendations to the users.

Recommender systems have been developed using data mining techniques [16,6,7,14,22,35,42,57], heuristics [39,11,46,50] and finding association patterns among the items [24,31]. Examples of popular recommender systems include Amazon.com [33] for books, CDs and various other products, Movie Lens [36] for movies, VERSIFI [5] for news, PHOAKS system for relevant information to users on web [55] and Jester system for jokes [16].

1.1. Motivation and problem definition

With the rapid growth of technology and the web, generation and accumulation of digital data have become easier. Advancement in techniques from diverse domains like machine learning, pattern recognition and statistics have made it possible to mine and unfold interesting as well as unknown patterns from the data.

Web data exists in various formats like url visits, web page content and incoming and outgoing hyperlinks to the page. Based on the data being analyzed or mined, web mining can be classified into three

[*] Corresponding author at: Faculty Block B, First Floor, IIM Indore, Indore, (M.P.), India 453331. Tel.: +91 731 2439 550; fax: +91 731 2439661.
E-mail addresses: rajhansm@iimidr.ac.in (R. Mishra), pradeepkumar@iiml.ac.in (P. Kumar), bhasker@iiml.ac.in (B. Bhasker).

http://dx.doi.org/10.1016/j.dss.2015.04.004
0167-9236/© 2015 Elsevier B.V. All rights reserved.

different categories, namely, web usage mining, web content mining and web structure mining. Web content mining is the application of data mining techniques to the content published on the web. Web structure mining operates on the web's hyperlink structure [56,13]. Web usage mining derives novel, implicit and useful patterns from the usage data of users [10,54,58].

Web recommendation systems are an important and popular tool to analyze users' behavior over the web and to generate recommendations as per their preferences. It supports organizations in intelligent decision making with respect to their customers' needs by automating the recommendations as per their preferences. Hence, it works as a decision support system for organizations.

Presenting a web user with his most probable next page visit is an interesting and challenging problem. Consider a web user who has registered with an online megastore website like ebay.com. In his session visit, he has traversed the pages like entertainment, books, electronics, footwear and so on. As a store manager, it will be an interesting problem to provide in advance to the web user the set of few web pages he/she might likely visit in his/her current session. Thus, any system providing a recommendation for the next one or two pages can be helpful in projecting the desired product or category to the user. Desired product/ category will have more probability to be purchased, which, in turn, can improve expected profits for any online e-commerce firm.

While building recommender systems, the sequentiality aspect of the user session is ignored. Sequential aspect of web user sessions has been considered by probabilistic models like Markov model while designing a web recommendation system. However, problem with the probabilistic model is that switching probability among web categories should be known a priori, and may require the knowledge and experience of a domain expert. Even if a domain expert is available, the estimation of exact probability among states (web categories/pages) is an open problem and cannot be easily addressed.

Recommendation system is a type of decision support system designed to discover user preferences, and to study them in order to anticipate their needs. They provide recommendations to customers as per their taste within a given domain. Formally, in a recommendation framework, there exists a large number (n) of items or products $P = \{P_1, P_2, P_3, \ldots, P_n\}$, which are described by a set of k attributes or features, $F = \{F_1, F_2, F_3, \ldots, F_k\}$. Each product is defined by one or more features from the feature set. There is also a large set of m users, $U = \{U_1, U_2, U_3, \ldots, U_m\}$ and for each user, a set of ratings about the quality of observed products is maintained in the database. Now, we formally define the problem as follows: For a new user p, the task of the system is to generate the set of next web page visit, based on the web page visits of similar profile available in database U. While predicting, the system should also consider the web page visit order.

Design of recommender system may be viewed as a combination of clustering and classification tasks. In this paper we have proposed the framework for the design of a recommender system using a combination of similarity upper approximation technique (for clustering web user sessions) and singular value decomposition (for predicting the next web page visit) algorithm [17,37]. To capture the sequentiality property of the data, we have used S^3M similarity measure [28] while performing clustering task.

1.2. Contribution and paper organization

In this paper, we designed a recommendation system for web users considering the sequential aspect of a web user session. The proposed recommendation system is different from sequential pattern mining algorithms. Sequential mining algorithms provide the patterns that exist in the sequences. In our work we have proposed a system which generates the recommendations to the users, considering the sequential information that exist in their usage patterns of web pages.

In our proposed model, rough set based similarity upper approximation clustering technique has been used that generates overlapping clusters. Overlapping clusters contain common elements, hence the boundary of these clusters become soft. Soft clusters are desired since they capture multiple interests of the users. They allow any user to be placed in multiple categories.

We have performed experiments to validate the results of our recommendation system. We have utilized three datasets for the experiments, the MSNBC dataset, simulated dataset and CTI dataset. We have evaluated the performance of our recommendation system on these three datasets and validated our results with the first order Markov model as well as a random prediction model.

The rest of the paper is organized as follows; related work has been discussed in Section 2. The architecture of the proposed system has been discussed in Section 3. Section 4 reports the experimental results and discussions while the conclusion and future work are given in Section 5.

2. Related work

The journey of recommenders system started with research papers on collaborative filtering by Resnick et al. [46], Shardanand and Maes [50] and Hill et al.[21]. Recommendation systems are designed using various techniques including k-NN, decision tree, clustering, regression, heuristic methods, neural networks and association rule mining [41]. Based on the type of techniques used, recommendation systems can be classified as content based and collaborative based systems [1].

The content based approach has originated from the information retrieval [46,48] and information filtering domain [36]. Content based recommender systems generate recommendations based on users' past preferences. The rating for any item for any user is calculated based on ratings of similar items given by the user. Many researchers treat it as a classification problem where the goal is to learn a function that predicts which class a document belongs to (i.e., either liked or not-liked). Others view it as a regression problem in which the goal is to learn a function that predicts a numeric value (i.e., the rating of the document) [4,30,38,43,32].

Collaborative systems are different from content based systems in the sense that they first find similar users for target users and then generate recommendations based on preferences of similar users [1]. In this approach, recommendations are made by finding correlations among the users. The main objective of collaborative filtering is to find rating of the items, not seen by the current user, using the ratings of similar users.

GroupLens [26], Video Recommender [21] and Ringo [50] are examples of recommender systems that use a collaborative filtering algorithm for automatic prediction. Collaboration based recommender systems can be further classified into two classes, memory based (heuristic based) and model based collaborative systems [1].

Memory based systems compute the similarity between users based on users' ratings. The algorithms of memory based systems are heuristics that make recommendations based on an entire collection of items pre-rated by the users [7,39,11,46,50]. Model based collaborative recommender systems generate the descriptive model of the system, based on the users' preferences, using various data mining and machine learning techniques. The techniques that are used include Bayesian models, clustering models, latent semantic models as singular value decomposition, probabilistic latent semantic analysis, multiple multiplicative factor, latent Dirichlet allocation and Markov decision process based models [59]. The predictions for a new user are made based on the constructed model. Kumar et al. [29] have used a simple probabilistic model for collaborative filtering. There are various other probabilistic modeling techniques used for building recommender systems, available in literature.

Zang et al. [60] have proposed a knowledge based recommender system that utilizes opinion mining and rough set association rule mining to find out the associations between product attributes from user data. Castellano et al.[8] have proposed a new system for web recommendation using fuzzy sets and neural networks. Other web page recommendation systems also exist in literature [40,3,52,15]. Decision support systems were developed for various important applications such as stock investment [18] and medicine [61]. Shani et al. [51] have proposed a system considering the recommendation process as a sequential decision process and utilized Markov's decision processes for generating recommendations. The Markov model happens to be a complex probabilistic model widely used for modeling sequential events [1]. Several hybrid recommendation systems have also been developed using multiple techniques including a sequential pattern analysis [23,34,9,47]. Most of the work related to the design of a recommendation system with sequential information used the Markov model [51,25,45].

Sequential and association pattern mining algorithms have been developed to find the sequential patterns in the data. These algorithms try to find associations among the items that exist in data points. AprioriAll [2,53] and PrefixSpan [44] have been the basic approaches to find sequential patterns. Sequential information embedded in the data is an important aspect that may be explored in various applications. In this work, a recommendation system that explores the sequential information present in data for generating recommendations has been developed.

Designing a recommendation system that considers sequential information is still an important problem that needs to be addressed. This type of recommendation system will help e-commerce websites develop a decision support system capable of capturing sequential information. In this work we have utilized the S^3M measure, which considers both the content as well as sequence of a visit during clustering to form groups of users. It is possible that a user may fall into more than one categories. A good recommendation system should be able to capture this information while forming the cluster. In order to capture the same information we have used a rough set based clustering using similarity upper approximation.

The proposed model uses a combination of clustering and classification techniques to generate recommendations considering sequential information. The clustering technique helps the system to group similar user profiles and the classifier learn the model from similar users to generate recommendations. Thus, the proposed recommendation system is a collaborative-model based system.

3. Proposed recommender system architecture

Generally, a pattern recognition based recommender system consists of two phases; the first phase is clustering followed by classification task. In the first phase, the system is provided with enough learning so that the classification accuracy of the system is quite high or at the desired level. After the system learns, it generates a set of recommendations with appropriate rankings.

In our system, we first formed clusters to acquire knowledge about web users and the classification technique was used later for enhancing the learning capability and to generate recommendations. A web user may have multiple interests for which he needs to be put into multiple clusters. Hence, we have used a similarity upper approximation based clustering algorithm. In order to capture sequential behavior of the user, we utilized S^3M [28] similarity measure while forming clusters. Soft clusters allow elements to appear in more than one cluster. This means a data point can represent the attributes of more than one cluster. Once the clusters are formed, we utilized the singular valued decomposition to classify the web user sessions. In Fig. 1, we have outlined the general architecture of the system. The first step is the collection of web data through web logs. After collecting web logs, the pre-processing is done, followed by the clustering stage. In the clustering module, each sequence is considered as a data point and all the points are clustered into several groups using a rough set based clustering algorithm that generates soft clusters allowing multiple interests of the users. After clustering, for any new user for whom recommendation has to be generated, Top M clusters are identified based on the similarity between user and cluster centers. A response matrix is created with the top M clusters. The details of creating the response matrix are explained in Section 3.2. After constructing the response matrix, a weight vector is created, which has been filled using singular value decomposition. This step is illustrated in Section 3.3. Generated predictions are compared with the original values of the datasets (test datasets) to evaluate the accuracy of the prediction. A detailed illustration of the proposed system is presented in Sections 3.1, 3.2 and 3.3.

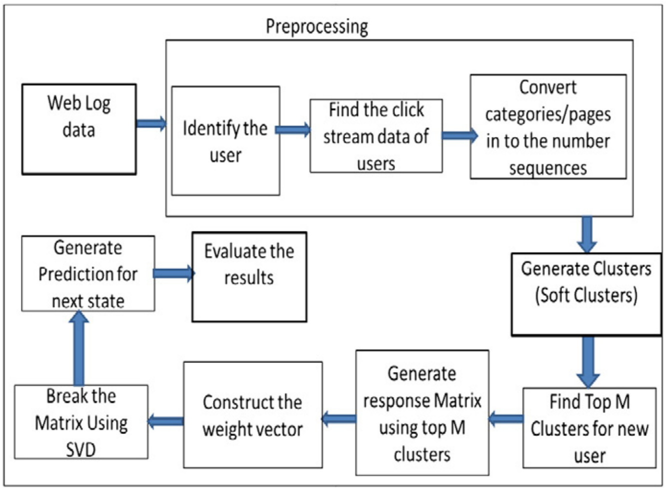
Preprocessing
Web Log data
Identify the user
Find the click stream data of users
Convert categories/pages in to the number sequences
Generate Clusters (Soft Clusters)
Find Top M Clusters for new user
Generate response Matrix using top M clusters
Construct the weight vector
Break the Matrix Using SVD
Generate Prediction for next state
Evaluate the results

Fig. 1. Proposed recommender system.

3.1. Clustering of web user sessions

The users are grouped based on a similarity measure using a clustering algorithm. We have utilized a rough set based clustering algorithm for clustering. Clustering is performed based on the similarity present among users. Similarity measures are used to estimate the similarity between objects. Content based similarity measures estimate the content similarity among users while sequence similarity measures estimate the sequence similarity among users. Jaccard and Dice similarity measures are examples of content based similarity measures while the Levensthein Distance, Longest Common Sub Sequence (LCS) and Hamming Distance are examples of sequence based similarity/distance measures. The combination of content and sequence based similarity measures result in hybrid similarity measures that capture the content and sequence similarity present among users. In our work, we have utilized a hybrid similarity measure during the clustering so content similarity and sequence similarity both are considered during clustering. S^3M [28] is a similarity measure which is a linear combination of the Jaccard [28] similarity measure and sequence similarity measure that is measured by the length of longest common subsequence (LLCS). We have utilized the similarity upper approximation of the rough set theory to come up with incremental soft clusters [27].

Let *UN* be the collection of web user sessions (representing universe set) and a non-empty set containing n user sessions denoted as $\{x_1, x_2, x_3, \ldots, x_n\}$. Each user session comprises of web page visits. Let D be a similarity matrix $[D]_{ij} = \mu(x_i, x_j)$, denoting the similarity among web user sessions x_i and x_j. The similarity between two web user sessions is computed using the S^3M measure [28]. The S^3M measure considers both the content as well as order (sequence) of the information while computing similarities between web user sessions. Once the similarity matrix has been computed, the initial set of clusters are formed using similarity upper approximation [27].

From the generated set of cluster family thus formed, only one set will be taken if two sets A and B are equal (where set A and set B are two sets out of several generated sets). Also, if set A is the proper subset of set B, then consider only set B. Thus, considering only unique and proper supersets from the cluster family set formed, a new set family is generated with reduced size.

However, the family of cluster sets is likely to be a pseudo-partition due to common elements in different sets. In order to have a natural grouping, it is necessary to partition the universe. In such a partitioning, an element should be in only one partition.

After forming the cluster, due to the first similarity upper approximation, a web user session will be a member of more than one group. Such objects are referred to as ambiguous objects. These collections of ambiguous objects form the soft clusters. The lower approximation of the set is the collection of web users that surely belong to a cluster.

We have outlined the algorithm for forming clusters from sequential data using similarity upper approximation as below:

```
Begin
    Step1: For two web user sessions x1 and x2 $\in T_D$ call function Sim (x1,x2)
    Step2: For a given $\delta \in (0, 1]$ form the first similarity upper approximation.
    Step3: Remove proper subsets.
    Step4: Identify the cluster centres, if exists, and remove them from other clusters.
End
Function Sim (x1, x2)
Input: p, x1 and x2 where $0 \le p \le 1$ Output: D a similarity matrix
Begin
    Step1: Compute SeqSim
        For $i^{th}$ element of x1 & $j^{th}$ element of x2
        LCS (i, j)
            $= \{ 0$ if $i = 0$ or $j = 0$;
            $= \{ \max |LCS(i-1, j), LCS(i, j-1)|$ if $xi \neq yj$&$i, j > 0$
            $= \{|LCS(i-1, j-1)| + 1$ if $xi = yj$&$i, j > 0$
$SeqSim(x1, x2) = \frac{LLCS}{\max(|x_1|, |x_2|}$
    Step2: Compute SetSim
$SetSim(x1, x2) = \frac{|x_1 \cap x_2|}{|x_1 \cup x_2|}$
    Step3: Compute $S^3M$
$S^3M = p \times SeqSim(x1, x2) + (1\text{-}p) \times SetSim(x1, x2)$
End
```

The illustration of the clustering technique used has been presented here. To explain the approach, consider 10 web user navigation patterns (S1, S2, S3, S4, S5, S6, S7, S8, S9 and S10) that have been taken from MSNBC dataset [27]. Similarity table between the sequences is computed using S^3M similarity metric with $p = 0.7$ (Table 1). "p" is the weight assigned by the domain expert, based on the sequential behavior present in the dataset. The process of computing S^3M between two sequences is illustrated below: Consider two sequesnce S1 and S2 representing the navigation patterns of two users of length L1 and L2.

$S1 = 1, 3, 5, 7, 9, 12$
$S2 = 3, 7, 12, 13, 15, 18, 19, 20$
$L1 = |S1| = 6$
$L2 = |S2| = 8$
SeqSim between S1 & S2 $= \frac{LLCS}{Max(L1, L2)}$
$LLCS = 3$, Max $(L1, L2) = 8$, SeqSim $= (3/8) = 0.375$
$S^3M = p \times SeqSim + q \times SetSim$
SetSim (Jaccard Similarity) between S1 & S2 $= \frac{|S1 \cap S2|}{|S1 \cup S2|}$
$|S1 \cap S2| = 3$, $|S1 \cup S2| = 11$, SetSim $= (3/11) = 0.272$
Considering, $p = 0.9$, $q = 0.1$
Thus, $S^3M = 0.3647$

Table 1
Similarity matrix for the sequences.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	1	0	0	0	0.21	0.29	0	0	0	0
S2	0	1	0	0.47	0.17	0.17	0.17	0	0.15	0.15
S3	0	0	1	0	0	0.25	0.33	0.33	0	0.21
S4	0	0.47	0	1	0.17	0	0.45	0.27	0.24	0.5
S5	0.21	0.17	0	0.17	1	0.18	0	0	0	0
S6	0.29	0.17	0.25	0	0.18	1	0.18	0.21	0	0.17
S7	0	0.17	0.33	0.45	0	0.18	1	0.58	0.17	0.62
S8	0	0	0.33	0.27	0	0.21	0.58	1	0	0.5
S9	0	0.15	0	0.24	0	0	0.17	0	1	0.24
S10	0	0.15	0.21	0.5	0	0.17	0.62	0.5	0.24	1

Step 1: As illustrated above, the computation of similarity between the 10 user sequences has been reported in the similarity table (see Table 1).

S1: on-air miscmiscmisc on-air misc
S2: news sports tech local sports sports
S3: bbs bbs bbs bbs bbs bbs
S4:frontpage frontpage sports news news local
S5:on-air weather weatherweatherweather sports
S6: on-air on-airon-airon-air tech bbs
S7: frontpagebbsbbsfrontpagefrontpage news
S8:frontpage frontpage frontpage frontpage frontpagebbs
S9: news news travel opinion opinion msn-news
S10:frontpage business frontpage news newsbbs

In Table 1 diagonal entries are equal to "1" as they represent the comparison of any sequence to itself. Other entries show the similarity between the different sequences.

Step 2: Assuming the similarity threshold $\delta = 0.2$, the upper approximation will be given as follows: R(1) = S1, S5, S6; R(2) = S2, S4; R(3) = S3, S6, S7, S8, S10; R(4) = S2, S4, S7, S8 , S9, S10; R(5) = S1, S5; R(6) = S1, S3, S6, S8; R(7) = S3, S4, S7, S8, S10; R(8) = S3, S4, S6, S7, S8, S10; R(9) = S4, S9, S10; R(10) = S3, S4, S7, S8, S9, S10.

Step 3: In the first upper approximation, a number of subsets also exist (that are redundant). Hence in the next step subsets are removed which will give the following set of expressions: R(1) = S1, S5, S6; R(4) = S2, S4, S7, S8 , S9, S10; R(6) = S1, S3, S6, S8; R(8) = S3, S4, S6, S7, S8, S10; R(10) = S3, S4,S7, S8, S9, S10.

After Step 3 only proper supersets will remain. Here S1, S4, S6, S8, S10 are the cluster centers.

Step 4: In this step, cluster centers will be removed from other clusters. After the removal of cluster centers, the set of expressions will be {{**1**, 5}, {**4**, 2, 7, 9},{**6**, 3},{**8**, 3, 7},{**10**,3, 7, 9}}. Here bold numbers represent cluster centers. These clusters are soft in nature as some of the objects (sequences) are present in more than one cluster.

3.2. Classification of web user sessions

After generation of clusters, it is utilized to generate the response matrix for the new user. This step has been illustrated in the current section considering the MSNBC dataset. The average length of web user session is 5.7 hence we have used only those sessions whose length is "6" for our experimentation purpose. Similarity between a new user and cluster centers has been calculated using their sequential pattern of length "5", and our recommender system will predict the sixth web page visit. Top "*M*" similar clusters will be selected for the formation of response matrix "A". In this case, each row of matrix *"A"* will contain seventeen columns (since there are 17 categories in the "MSNBC" dataset).

The row vector of response matrix *"A"* corresponding to the first cluster is a_1. *M*th cluster will be represented by vector a_m. Pages that

Table 2
Sample response matrix (A).

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
a1	5	0	0	0	2	0	0	6	3	7	0	0	0	0	1	4	5
a2	4	11	3	6	4	5	7	1	8	9	5	9	5	8	3	23	5
a3	3	4	4	5	8	0	3	3	5	4	6	7	13	0	6	4	3
a4	6	18	8	5	4	6	4	13	4	4	8	0	2	7	12	2	1
a5	8	0	9	2	9	4	3	4	7	0	7	0	16	6	14	6	6
a6	9	3	3	5	1	2	0	1	2	1	7	11	4	11	1	7	4
a7	0	6	2	9	11	6	7	7	3	7	7	3	7	22	8	2	3
a8	0	1	6	5	5	0	12	5	15	8	2	8	12	4	9	9	5
a9	6	7	8	0	4	1	8	0	12	4	1	6	3	2	3	2	3
a10	2	3	2	2	2	5	5	4	3	3	5	3	1	0	1	0	2
a11	4	0	0	5	7	8	9	0	6	2	7	2	8	9	5	8	1
a12	1	18	2	0	12	7	9	9	2	6	9	5	13	11	4	3	9
a13	0	8	4	3	1	12	13	6	0	7	7	0	4	7	3	2	4
a14	2	4	33	3	9	8	14	8	4	8	2	7	3	6	1	6	2
a15	5	5	12	6	0	5	2	4	8	9	2	8	12	1	7	2	3
a16	2	0	1	0	5	8	11	13	2	3	13	4	2	4	2	8	5
a17	0	0	23	2	8	9	0	3	4	13	14	6	1	22	19	1	1
a18	8	4	7	8	11	7	0	2	0	4	6	0	6	21	2	6	9
a19	2	1	3	9	2	4	9	12	0	17	1	7	4	11	3	3	12
a20	6	1	23	1	9	7	7	6	0	19	8	8	3	9	7	8	0

are not present as the 6th state of any member in the cluster will be represented with "0". Frequency of occurrence of web pages has been reported in vector a_1. It is represented as $a_1 = \{5,0,0,0,2,0,0,6,3,7,0,0,0,0,1,4,5\}$ (Table 2).

Top *M* similar clusters will generate *M* rows of matrix *A* and will be represented as a_1, a_2, a_3,…,a_m. Matrix *"A"* will be a matrix of size $M \times T$, where *T* represents the total categories available.

In the current example, *T* happens to be 17. a_2, a_3,…,a_m can also be generated in the same manner as a_1. Let $M = 20$, then the size of Matrix *"A"* will be 20×17. The response matrix (A) has been shown in Table 2 for the purpose of illustration. Response matrix (A) is formed by the Top M similar clusters and is a collection of row vectors a_1, a_2, a_3,…,a_m. Matrices *U, S,* V^T will be generated using singular value decomposition (SVD). The size of *U* will be 20×17, the size of *S* will be 17×17 and the size of V^T will be 17×17. Diagonal elements of matrix *S* will be non-zero, other elements will be zero.

3.3. Recommendations for the web user

The next step after the construction of response matrix "A" is to construct a weight vector for new users. This step is illustrated in this section. Let the sequential pattern of a user for the first five visits be {3, 8, 7, 5, 1}. The weight of any page or category i visited by the user in jth position has been termed as Wij and can be calculated as per Eq. (1).

$$\text{Wij} = \frac{\left|V_{ij}\right|}{|V_i|}, \tag{1}$$

where, |Vij| = number of times page i has been present in jth position.
|Vi| = number of times page i has been present in all positions.
The calculation of Wij is explained as follows:
Let the T1, T2, T3, T4 be four sequences of length "6".
T1 = 3, 7, 8, 3, 1, 9; T2 = 5, 8, 3, 5, 4, 8; T3 = 7, 1, 8, 13, 2, 6; T4 = 5, 15, 7, 13, 2, 6.

Let the sequential page visits of the new user be {3, 8, 7, 5, 1}. The weights of the pages appearing at different places can be calculated as follows:

$\text{W31} = \frac{|1|}{|3|} = 0.33$, $\text{W82} = \frac{|1|}{|4|} = 0.25$, $\text{W73} = \frac{|1|}{|3|} = 0.33$, $\text{W54} = \frac{|1|}{|3|} = 0.33$, $\text{W15} = \frac{|1|}{|2|} = 0.50$.

For next page visit, the weight Wk6 has to be calculated where, k = {1, 2, 3,…,17}. A weight vector P1 of length "17" will be formed for the new user. The entries of this vector will be the weights of the corresponding page/category. Initially P1 = {0.5,X,0.33,X,0.33,X,0.33,0.25,X,X,X,X,X,X,X,X,X}. The unknown entries (represented by 'X') of P1 will be calculated as follows:

Let the weight of the dth page be R_d and U, S, V represent the decomposed matrices of response matrix "A", "i" represents the ith user and "k" represents the kth feature. R_d can be represented by Eq. (2).

$$R_d = \sum\nolimits_k U_{ik} S_{kk} V_{jk} \tag{2}$$

In this case R1 = 0.50, R3 = 0.33, R5 = 0.33, R7 = 0.33 and R8 = 0.25 are known for the new user. The mathematical formulation has been shown by Eqs. (3)–(7).

$$R_3 = U_3 S_{33} V_{33} + U_8 S_{88} V_{38} + U_7 S_{77} V_{37} + U_5 S_{55} V_{35} + U_1 S_{11} V_{31} \tag{3}$$

$$R_8 = U_3 S_{33} V_{83} + U_8 S_{88} V_{88} + U_7 S_{77} V_{87} + U_5 S_{55} V_{85} + U_1 S_{11} V_{81} \tag{4}$$

$$R_7 = U_3 S_{33} V_{73} + U_8 S_{88} V_{78} + U_7 S_{77} V_{77} + U_5 S_{55} V_{75} + U_1 S_{11} V_{71} \tag{5}$$

$$R_5 = U_3 S_{33} V_{53} + U_8 S_{88} V_{58} + U_7 S_{77} V_{57} + U_5 S_{55} V_{55} + U_1 S_{11} V_{51} \tag{6}$$

$$R_1 = U_3 S_{33} V_{13} + U_8 S_{88} V_{18} + U_7 S_{77} V_{17} + U_5 S_{55} V_{15} + U_1 S_{11} V_{11} \tag{7}$$

Values of U_3, U_8, U_7, U_5, U_1 will be calculated by solving Eqs. (3)–(7) simultaneously. The weight of any category "d" can be calculated using Eq. (8) given below.

Table 3
Weights of different pages for next step.

Page	Weight	Suggestion	Page	Weight	Suggestion
R1	0.50	9	R10	1.90	3
R2	0	17	R11	1.46	4
R3	0.33	11	R12	1.11	5
R4	0.80	7	R13	0	16
R5	0.33	12	R14	2.53	1
R6	0.92	6	R15	0.69	8
R7	0.33	13	R16	2.20	2
R8	0.25	14	R17	0.49	10
R9	0.06	15			

$$R_d = U_3 S_{33} V_{d3} + U_8 S_{88} V_{d8} + U_7 S_{77} V_{d7} + U_5 S_{55} V_{d5} + U_1 S_{11} V_{d1} \tag{8}$$

The page with highest weight among the new ranked pages will be the next suggested page. Sequence of pages can be suggested based on the ratings of new pages. The weights of different page categories have been reported in Table 3 after calculation. It is clear from Table 3 that page category R14 happens to be most preferred and recommended as the next page visit of the new user.

The algorithmic view of the proposed model has been represented below:

Algorithm

```
Input
    Dataset containing sequence of web log data = $|U_D|$
    Sequential visits of new user = n
    Number of top clusters = M
Output
    Recommendations for new user = R
Begin
    Step1: Identify different users.
    Step2: Compile click stream data of users into a single sequence
  (Each sequence will represent a user) = $|U|$.
    Step3: Apply clustering algorithm to generate clusters considering sequential similarity.
    Step4: Find the top-M similar clusters for new user n.
    Step5: Construct the Response Matrix for new user n.
    Step6: Construct the weight vector for new user considering the location of various pages.
    Step7: Apply Singular Value Decomposition (SVD) on Response Matrix to break the matrix.
    Step8: Apply prediction function of SVD to generate ratings of web pages for new user.
    Step 9: Return the set of recommendations R.
End
```

4. Experimental results and discussion

We have performed our experiments on three datasets. The first data set is the MSNBC web navigation data set, which is a bench mark dataset. The MSNBC web navigational dataset has been collected from the UCI dataset repository. The dataset consists of Internet Information Server (IIS) logs for msnbc.com web site and news-related portions of msn.com for the entire day of September 28, 1999 (Pacific Standard Time). Each web log is a sequence representation of page views of a web user during the 24 hour period. The length of the web user session varies from 1 to 500. The average length of the web user session is reported at 5.7, hence for our experiments we have taken only those user sessions whose length is six. The data set has seventeen categories: “frontpage”, “news”, “tech”, “local”, “opinion”, “on-air”, “misc”, “weather”, “health”, “living”, “business”, “sports”, “summary”, “bbs” (bulletin board service), “travel”, “msn-news” and “msn-sports”. We have converted the categories into numbers starting from 1 to 17 for the purpose of experiments.

We have also prepared a simulated dataset and performed our experiments on it. The simulated data set has 25 categories that have been numbered from 1 to 25. The length of each sequence is 13. These sequences have been generated using the random generator function in Microsoft Excel.

Experiments were also carried on CTI dataset. CTI dataset has been collected from the DePaul CTI Web server (http://www.cs.depaul.edu). It is based on a random sample of users visiting this site for a 2 week period during April of 2002. The original (unfiltered) data contained 20,950 sessions from 5446 users. The filtered data files were produced by filtering low support page views and eliminating sessions of size 1. The filtered data contains 13,745 sessions and 683 page views. We have clubbed the page views into the original root of that page (categories), resulting the total number of web categories to 15.

Several metrics exist in literature for the evaluation of recommendation systems [19,20]. The most important metric is accuracy. There are other metrics, such as precision, recall and coverage. We have performed experiments to estimate the accuracy of the proposed recommendation system and also reported precision of the system.

Accuracy: Accuracy has been defined as the ratio of the number of correct recommendations to the number of total recommendations.

$$\text{Accuracy} = (\text{Number of correct recommendations})/(\text{Number of total recommendations}) \tag{9}$$

In this case, we recommend the next possible visit of the user, which is compared with the actual next step of the user (testing). If the predicted next step is the same of the actual next state, then the event is termed as hit, else it is termed as miss. Hence, accuracy is our case will be given as:

$$\text{Accuracy} = (\text{Total number of hits})/(\text{Total number of hits} + \text{Total number of miss}) \tag{10}$$

Considering the confusion matrix (Table 4), Accuracy of the recommendation system can be defined as the ratio of relevant retrieved elements to all retrieved, non-retrieved, relevant and non relevant elements.

$$\text{Accuracy} = a/(a + b + c + d) \tag{11}$$

Precision [29]. Precision is a metrics of information retrieval domain and is considered as an important parameter of recommendation systems along with accuracy. It uses a matrix, termed as a confusion matrix (Table 4) which represents the relevant items among all retrieved items.

Precision has been defined ratio of relevant retrieved element to all retrieved elements.

$$\text{Precision} = a/(a + b) \tag{12}$$

In the proposed recommendation system, recommendations are generated based on a mathematical model and computation, hence it gives a recommendation for all the inputs (the recommendation may be correct or incorrect). Due to this reason there will not be any non-retrieved entry, hence $c = 0$, $d = 0$ [59].

4.1. Experimental results

In this subsection, we have reported the experimental results. The proposed recommendation system uses the user defined parameter *M* (number of clusters chosen for constructing the response matrix *A*). The output may vary with the choice of the value of *M*. Experiments were carried for different values of *M*. We have taken 5000 sequences for clustering and 10 different groups from a data set of size 2000 of *MSNBC* web navigation dataset for testing. We performed a ten-fold cross validation of our proposed model.

Similar experiments have been performed with the simulated dataset where 5000 sequences have been taken for the clustering and

Table 4
Sample confusion matrix.

	Relevant	Non-relevant
Retrieved	a	b
Not retrieved	c	d

Table 5
Representation of percentage prediction accuracy of first five samples using S^3M similarity considering soft clustering with $M = 20$ (for the MSNBC dataset).

No. of predictions	Random	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
1	5.88	11.76	12.50	9.38	17.24	13.64
2	11.76	23.53	18.75	18.75	24.14	13.64
3	17.65	29.41	37.50	28.13	31.03	22.73

Table 7
Representation of percentage prediction accuracy of first five samples using S^3M similarity considering soft clustering with $M = 20$ (for CTI dataset).

No. of predictions	Random	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
1	6.25	9.38	6.90	7.41	6.90	6.25
2	12.50	21.88	31.03	18.52	48.28	31.25
3	18.75	28.13	34.48	25.93	48.28	37.50

10 sets of dataset of 500 have been taken for testing. We performed a 10-fold cross validation for the simulated dataset and compared the accuracy of the prediction with random predictions as well as with first order Markov model. Similar experiments have been performed with CTI datasets.

We have validated the results of the proposed model with the random prediction model and first order Markov based model. We utilized a 10-fold cross validation in which the total test sample is divided into 10 sub-samples to test the robustness of the results. We performed the experiments on three datasets (MSNBC, simulated and CTI) and the results are reported in this sub-section. Table 5 shows the accuracy of the proposed model for various samples of the MSNBC dataset. It also reports the accuracy of the random prediction model for the same samples. It is clear from Table 5 that the accuracy of the proposed model is more than the accuracy of the random prediction model. Table 6 represents the accuracy of the proposed recommendation system and its comparison with the random prediction model for various samples of the simulated dataset. The same facts have been shown for the CTI dataset in Table 7. It is clear from Tables 5–7 that accuracy of the proposed model has outperformed the accuracy of random prediction model for various samples of all the three datasets (MSNBC, simulated, CTI) used for the experiments.

4.2. Validation with the Markov model

The Markov model considers the sequential patterns of the user to predict the next state of the user. We have compared our recommendation system with the first order Markov model based system for validation.

The page visits of any user forms the browsing history of that user. The web site visits of any user forms a user session. A user's web site visits can be modeled by observing the browsing history [54]. Markov based model also considers the sequential information for predicting the next state. They compute the switching probability among the different states, which is utilized to predict the probability of various states as the next state. The state with the highest probability has been recommended as next state by the Markov model based system [12]. The user's web session W(set of pages visited by user) can be represented as sequence of web pages [59] where W = <S_1, S_2, …, S_l>, S_1 represents the first page visited by user, S_2 represents the second page visited by user and S_l represents the lth page visited by the user. S_T represents a set of all possible web pages, hence “W” happens to be a subset of S_T . Let W be a web session of a user, comprised of l pages. The prediction of the next page can be modeled using the probabilistic framework. Let P (s_i/W) be the probability that the user will visit page “i” in the next page visit whose web session of length l is W. Then $s_{l\,+\,1}$ is the next page visit after the l page visits and is given as:

$$\begin{aligned} S_{l+1} &= \underset{p\in S}{\arg\max}\{P(S_{l+1} = s_i \mid W\} \\ &= \underset{p\in S}{\arg\max}\{P(S_{l+1} = s_i \mid S_l, S_{l-1}, \ldots, S_1\}. \end{aligned} \tag{13}$$

Table 6
Representation of percentage prediction accuracy of first five samples using S^3M similarity considering soft clustering with $M = 85$ (for simulated dataset).

No. of predictions	Random	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
1	4.00	6.15	8.70	6.49	5.97	10.00
2	8.00	9.23	11.59	12.99	8.96	15.71
3	12.00	18.46	15.94	16.88	16.42	20.00

It is clear from Eq. (13) that calculation has been done for the probability of each page (si) which can be the next page ($s_i \in S_T$). The page with the highest value of probability is to be recommended as the next page. Fig. 2 reports the average accuracy of the proposed recommendation model considering a 10-fold cross validation for various values of M and compares it with a random prediction model as well as first order Markov model. The same facts with respect to the simulated dataset and CTI dataset has been reported in Figs. 3 and 4. Results reported in Figs. 2–4 further test the robustness of the proposed model for various values of M for different datasets. The experiments have been performed with various values of M and the proposed system has performed better than the random prediction model and first order Markov model. The 10-fold cross validation further validates the performance of the proposed system as it outperforms first order Markov model and random prediction model for three datasets.

Though prediction accuracy happens to be the most important parameter to judge the performance of the recommendation system, we have also reported precision of the proposed model. Table 8 represents the precision for first page, first two pages and first three pages predictions for the MSNBC dataset, simulated and CTI dataset. It is clear from Table 8 that the proposed recommendation system has outperformed the random prediction model in terms of precision. Precision has been reported for different datasets for various values of

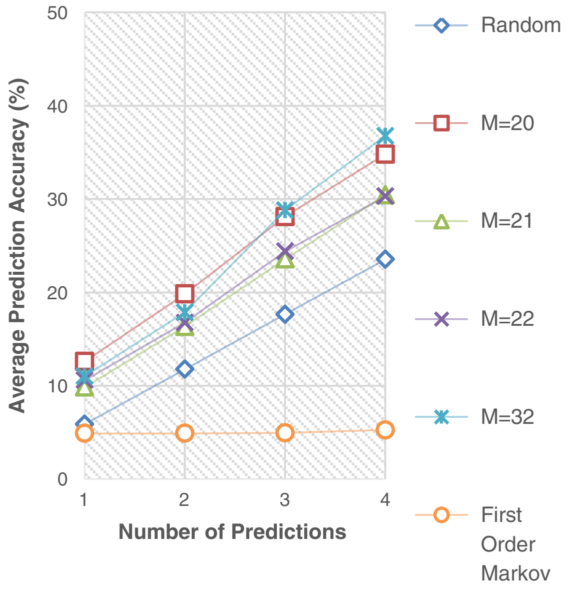

Fig. 2. Comparison of accuracy of the proposed model with the first order Markov model and random model for MSNBC dataset.

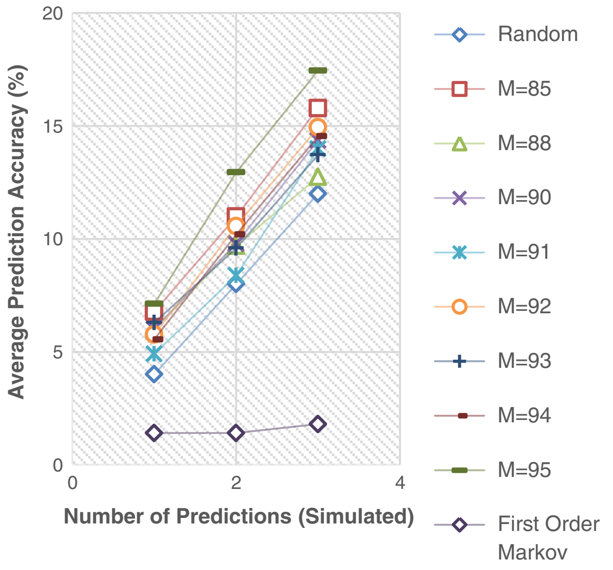

Fig. 3. Comparison of accuracy of the proposed model with the first order Markov model and random model for simulated dataset.

M. Table 8 reports the robustness of performance of the proposed model.

5. Conclusion and future work

With the accelerated growth in the number of web users, the web has become increasingly populated with data. Thus web based recommendation systems have gained significance. They work as intelligent systems to predict the behavior of users, hence help as decision support systems of organizations. Web recommender systems are quite popular in e-commerce applications. They can be utilized to judge the behavior of web users, based on their usage patterns. A user's behavior plays an important role for e-marketers to customize its offerings to a web customer.

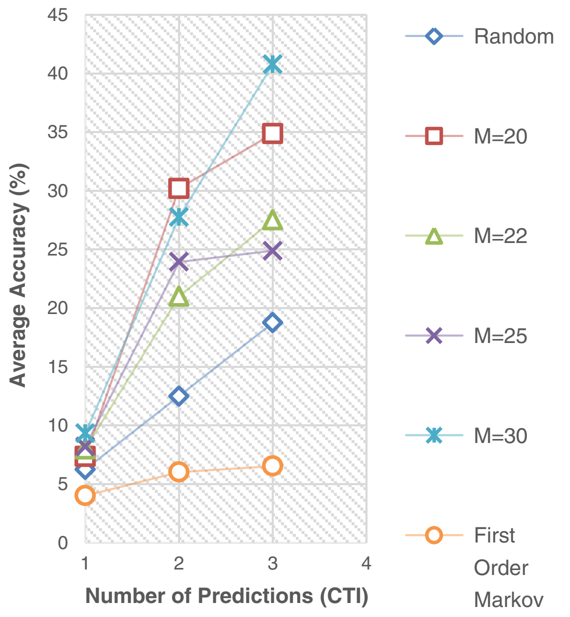

Fig. 4. Comparison of accuracy of the proposed model with the first order Markov model and random model for CTI dataset.

Table 8
Precision percentage for different page predictions.

	MSNBC dataset			Simulated dataset			CTI dataset		
	First	Second	Third	First	Second	Third	First	Second	Third
$M = 20$	12.61	19.83	28.12	6.78	10.98	15.79	7.37	30.19	34.86
$M = 22$	10.60	16.74	24.38	6.01	9.79	14.38	7.98	21.00	27.51
$M = 32$	11.02	17.86	28.80	7.12	12.95	17.46	8.20	23.94	24.89
Random	5.88	11.76	17.65	4	8	12	4	8	12
Markov model	4.90	4.90	4.95	1.40	1.80	1.80	4.02	6.03	6.53

In this paper, we have proposed a novel model to predict a user's next page visit. Judging the next page visit of a user can give useful information about their likes and taste. This information can be explored and utilized for projecting the desired category or product to the user. In this way, the probability of purchase for e-commerce organizations can be enhanced, which in turn will increase the expected revenue for e-commerce organizations.

We have used similarity upper approximation and singular valued decomposition for developing a novel recommendation system. A rough set based similarity upper approximation concept has been utilized during clustering which generates soft clusters. We have considered both sequential similarity and content similarity during clustering. We have utilized the S^3M similarity measure, which is a hybrid of content and sequential similarity measures. The generated soft clusters have been utilized to create a response matrix which is used by singular value decomposition to generate predictions. The proposed model is a novel model that considers sequential information for predicting a user's next visit. We have compared the results of our model with the random prediction and first order Markov based models. The experiments have been performed on three datasets, the MSNBC web navigation dataset, simulated dataset and CTI dataset. The results have proved the viability of our approach. In this work, we have tried to predict the next page visit of the user by using his sequential visits. We have not used information other than the navigation behavior of users.

Demographic variables can also be important for finding similar users. Similar users, using demographic variables, can be used as a reference for new users. Demographic variables will be more important in the case of a cold start problem when a users' navigation pattern is unknown. Hence, in future, a system can be developed that will use demographic information in addition to sequential navigation patterns. This system can be utilized to find similar users and predict the next step for new users; thus, it can address the cold start problem in a better way. In future an augmented system can be developed on the top of the current system considering demographic information, which will be able to address cold start problem and provide better personalization. It will result in an improved decision support system and help organizations to realize more revenue from the customers by effective web personalization using the recommendations of the developed system.

References

[1] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering 17 (6) (2005) 734–749.

[2] R. Agrawal, R. Srikant, Mining sequential patterns, Proceedings of the Eleventh International Conference on Data Engineering, IEEE 1995, pp. 3–14.

[3] M. Balabanović, An adaptive web page recommendation service, Proceedings of the First International Conference on Autonomous Agents, ACM 1997, pp. 378–385.

[4] M. Balabanovic, Y. Shoham, Fab: content-based, collaborative recommendation, Comm. ACM 40 (3) (1997) 66–72.

[5] D. Billsus, C.A. Brunk, C. Evans, B. Gladish, M. Pazzani, Adaptive interfaces for ubiquitous web access, Comm. ACM 45 (5) (2002) 34–38.

[6] D. Billsus, M.J. Pazzani, Learning collaborative information filters, 15th International Conference on Machine Learning, Madison, WI 1998, pp. 46–53.

[7] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, Proceedings of 14th Conf. Uncertainty in Artificial Intelligence, 1998 (July).

[8] G. Castellano, A.M. Fanelli, M.A. Torsello, NEWER: A system for NEuro-fuzzy WEb Recommendation, Applied Soft Computing 11 (1) (2011) 793–806.

[9] K. Choi, D. Yoo, G. Kim, Y. Suh, A hybrid online-product recommendation system: combining implicit rating-based collaborative filtering and sequential pattern analysis, Electronic Commerce Research and Applications 11 (4) (2012) 309–317.

[10] R. Cooley, B. Mobasher, J. Srivastava, Web mining: information and pattern discovery on the World Wide Web, Paper presented at the ninth IEEE international conference on tools with artificial intelligence, Newport Beach, CA, USA, 1997.

[11] J. Delgado, N. Ishii, Memory-based weighted-majority prediction for recommender systems, Proceedings of ACM SIGIR '99 Workshop Recommender Systems: Algorithms and Evaluation, 1999.

[12] M. Deshpande, G. Karypis, Selective Markov models for predicting Web page accesses, ACM Transactions on Internet Technology 4 (2) (2004) 163–184.

[13] F. Fu, X. Chen, L. Liu, L. Wang, Social dilemmas in an online social network, The Structure and Evolution of Cooperation Physics Letters 371 (2007) 58–64.

[14] L. Getoor, M. Sahami, Using probabilistic relational models for collaborative filtering, Proc. Workshop Web Usage Analysis and User Profiling (WEBKDD '99), 1999 (Aug.).

[15] M. Göksedef, S.G. Öğüdücü, Combination of web page recommender systems, Expert Systems with Applications 37 (4) (2010) 2911–2922.

[16] K. Goldberg, T. Roeder, D. Gupta, C. Perkins, Eigentaste: a constant time collaborative filtering algorithm, Information Retrieval Journal 4 (2) (2001) 133–151.

[17] G.H. Golub, C.F. van Loan, Matrix Computations, 3rd edition John Hopkins University Press, 1996.

[18] J. Gottschlich, O. Hinz, A decision support system for stock investment recommendations using collective wisdom, Decision Support Systems 59 (2014) 52–62.

[19] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, AACM Transactions on Information Systems (TOIS) 22 (1) (2004) 5–53.

[20] F. Hernández delOlmo, E. Gaudioso, Evaluation of recommender systems: a new approach, Xxpert Systems with Applications 35 (3) (2008) 790–804.

[21] W. Hill, L. Stead, M. Rosenstein, G. Furnas, Recommending and evaluating choices in a virtual community of use, The Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1995.

[22] T. Hofmann, Latent semantic models for collaborative filtering, ACM Transaction. Information Systems 22 (1) (2004) 89–115.

[23] Y.M. Huang, Y.H. Kuo, J.N. Chen, Y.L. Jeng, NP-miner: a real-time recommendation algorithm by using web usage mining, Knowledge-Based Systems 19 (4) (2006) 272–286.

[24] C.L. Huang, W.L. Huang, Handling sequential pattern decay: developing a two-stage collaborative recommender system, Elsevier, Electronic Commerce Research and Applications 8 (3) (2009) 117–129.

[25] Y.M. Huang, T.C. Huang, K.T. Wang, W.Y. Hwang, A Markov-based recommendation model for exploring the transfer of learning on the Web, Educational Technology & Society 12 (2) (2009) 144–162.

[26] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, J. Riedl, GroupLens: applying collaborative filtering to usenet news, Comm. ACM 40 (3) (1997) 77–87.

[27] P. Kumar, P.R. Krishna, R.S. Bapi, S.K. De, Clustering using similarity upper approximation, Proceedings of IEEE International Conference on Fuzzy Systems, 2006.

[28] P. Kumar, B.S. Raju, P.R. Krishna, A new similarity metric for sequential data, International Journal of Data Warehousing and Mining (IJDWM) 6 (4) (2010) 16–32.

[29] R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, Recommendation systems: a probabilistic analysis, Journal of Computer and System Sciences 63 (1) (2001) 42–61.

[30] K. Lang, Newsweeder: learning to filter netnews, Proc. 12th Int'l Conf. Machine Learning, 1995.

[31] C.H. Lee, M.S. Chen, C.R. Lin, Progressive partition miner, an efficient algorithm for mining general temporal association rules, IEEE Transactions on Knowledge and Data Engineering 15 (4) (2003) 1004–1101.

[32] T.P. Liang, Y.F. Yang, D.N. Chen, Y.C. Ku, A semantic-expansion approach to personalized knowledge recommendation, Decision Support Systems 45 (3) (2004) 401–412.

[33] G. Linden, B. Smith, J. York, Amazon.com recommendations: item-to-item collaborative filtering, IEEE Internet Computing, 2003.

[34] D.R. Liu, C.H. Lai, W.J. Lee, A hybrid of sequential rules and collaborative filtering for product recommendation, Information Sciences 179 (20) (2009) 3505–3519.

[35] B. Marlin, Modeling User Rating Profiles for Collaborative Filtering, Proc. 17th Ann. Conf. Neural Information Processing Systems (NIPS '03), 2003.

[36] B.N. Miller, I. Albert, S.K. Lam, J.A. Konstan, J. Riedl, MovieLens unplugged: experiences with an occasionally connected recommender system, Proc. Int'l Conf, Intelligent User Interfaces, 2003.

[37] R. Mishra, P. Kumar, B. Bhasker, A design framework for recommender system by incorporating sequential information, In Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services (iiWAS '11), 2011.

[38] R.L. Mooney, L. Roy, Content-based book recommending using learning for text categorization, Proc. ACM SIGIR '99 Workshop Recommender Systems: Algorithms and Evaluation, 1999.

[39] A. Nakamura, N. Abe, Collaborative filtering using weighted majority prediction algorithms, Proc. 15th Int'l Conf. Machine Learning, 1998.

[40] S. Niwa, S. Honiden, Web page recommender system based on folksonomy mining for ITNG &# 146; 06 submissions, Information Technology: New Generations, ITNG 2006, Third International Conference on (388-393). IEEE, 2006.

[41] D.H. Park, H.K. Kim, Y. Choi, J.K. Kim, A literature review and classification of recommender systems research, Expert Systems with Applications 39 (1) (2012) 10059–10072.

[42] D. Pavlov, D. Pennock, A Maximum Entropy Approach to Collaborative Filtering in Dynamic, Sparse, High-Dimensional Domains, Proc. 16th Ann. Conf. Neural Information Processing Systems (NIPS '02, 2002.

[43] M. Pazzani, D. Billsus, Learning and revising user profiles: the identification of interesting web sites, Machine Learning 27 (1997) 313–331.

[44] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, M.C. Hsu, Mining sequential patterns by pattern-growth: the prefix span approach, Knowledge and Data Engineering, IEEE Transactions on 16 (11) (2004) 1424–1440.

[45] S. Rendle, C. Freudenthaler, L. Schmidt-Thieme, Factorizing personalized Markov chains for next- basket recommendation, Proceedings of the 19th International Conference on World WideWeb, ACM 2010, pp. 811–820.

[46] P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, Proc. 1994 Computer Supported Cooperative Work Conference, 1994.

[47] M. Salehi, I.N. Kamalabadi, Hybrid recommendation approach for learning material based on sequential pattern of the accessed material and the learner's preference tree, Knowledge-Based Systems, 2013.

[48] G. Salton, Automatic Text Processing, Addison-Wesley, 1989.

[49] J.B. Schafer, J. Konstan, J. Riedl, Recommender systems in e-commerce, Proceedings of the 1st ACM conference on Electronic commerce, ACM 1999, pp. 158–166.

[50] U. Shardanand, P. Maes, Social information filtering: algorithms for automating 'word of mouth', Proc. Conf. Human Factors in Computing Systems, 1995.

[51] G. Shani, R. Brafman, D. Heckerman, G. Shani, R. Brafman, D. Heckerman, An MDP-based recommender system, Proc. 18th Conf. Uncertainty in Artificial Intelligence, 2002.

[52] S.K. Sharma, U. Suman, An efficient semantic clustering of URLs for web page recommendation, International Journal of Data Analysis Techniques and Strategies 5 (4) (2013) 339–358.

[53] R. Srikant, R. Agrawal, Mining sequential patterns: generalizations and performance improvements, Proceedings of the 5th International Conference on Extending Database Technology (EDBT), Avignon, France 1996, pp. 3–17.

[54] J. Srivastava, R. Cooley, M. Deshpande, P.N. Tan, Web usage mining: discovery and applications of usage patterns from web data, SIGKDD Explorations 1 (2) (2000) 12–23.

[55] L. Terveen, W. Hill, B. Amento, D. McDonald, J. Creter, PHOAKS: a system for sharing recommendations, Comm. ACM 40 (3) (1997) 59–62.

[56] I.H. Ting, I.H. Ting, Web Mining Techniques for On-line Social Networks Analysis., in the Proceedings of International Conference on Service Systems and Service Management, 2008.

[57] L.H. Ungar, D.P. Foster, Clustering Methods for Collaborative Filtering, in Proc. of Recommender Systems, Papers from Workshop, Technical Report WS-98-08, 1998.

[58] J. Wang, I. NetLibrary, Encyclopedia of Data Warehousing and Mining: Idea Group Reference, 2006.

[59] S. Xiaoyuan, T.M. Khoshgoftaar, A survey of collaborative filtering techniques, Advances in Artificial Intelligence archive, 2009.

[60] H. Zang, Y. Xu, Y. Li, Non-redundant sequential association rule mining and application in recommender systems, Web Intelligence and Intelligent Agent Technology (WI-IAT), IEEE/WIC/ACM International Conference 2010, pp. 292–295.

[61] Z.Y. Zhuang, C.L. Wilkin, A. Ceglowski, A framework for an intelligent decision support system: a case in pathology test ordering, Decision Support Systems 55 (2) (2013) 476–487.

Rajhans Mishra is an Assistant Professor in Information Systems Area at Indian Institute of Management Indore. He has also served as a visiting faculty at Indian Institute of Management Ahmedabad and Indian Institute of Management Lucknow. His research interest includes recommender systems, web mining, data mining, text mining, e-Governance and business analytics. He has completed his doctoral work from Indian Institute of Management Lucknow.

Dr. Pradeep Kumar is currently an Assistant Professor in IT and Systems area at IIM Lucknow. Prior to joining IIM, he has served SET Labs, Bangalore and Institute for Development and Research in Banking Technology (IDRBT), Hyderabad. He received his Ph.D. from Department of Computer and Information Sciences, Hyderabad University, India. He holds M.Tech. and B.Sc.(Engg.) in Computer Science. His research interest includes Data Mining, Web Mining, Soft Computing, Network Security and Image Processing.

Dr. Bharat Bhasker is a Professor in IT and Systems area at IIM Lucknow. He received his B. E. (Electronics and Comm. Engg.) from University of Roorkee and his M.S. and Ph.D. in Computer Science from Virginia Tech, USA. He has been honored with many prestigious awards. His research interests include Distributed Database Management, Data Mining, Personal Recommendation Systems, and Agent based Electronic Shopping. He has also authored a book on Electronic Commerce.