

CONSTRAINT INFORMATIVE RULES FOR GENETIC ALGORITHM-BASED WEB PAGE RECOMMENDATION SYSTEM

¹Prince Mary, S. and ²E. Baburaj

¹Department of Computer Science Engineering, Sathyabama University, Chennai-119, India

²Department of Computer Science Engineering,
Sun Engineering College, Nagercoil, Kanyakumari Dist-629902, India

Received 2013-06-20, Revised 2013-07-23; Accepted 2013-10-05

ABSTRACT

To predict the users navigation using web usage mining is the primary motto of the web page recommendation. Currently, researchers are trying to develop a web page recommendation using pattern mining technique. Here, we propose a technique for web page recommendation using genetic algorithm. It consists of three phases as data preparation, mining of informative rules and recommendation. The data preparation contains data preprocessing and user identification. The genetic algorithm is used to mine the informative rule. The genetic algorithm involves three processes which are calculating the fitness values, crossover and mutation. We use three different constraints as time duration, quality and recent visit to allow the process for next stage after the initial fitness calculation. We have to repeat these processes to find the best solution. To form the recommendation tree, we use the best solution which we obtain by means of genetic algorithm.

Keywords: Genetic Algorithm, Recommendation Tree, Constraint Rules, Web Page Recommendation

1. INTRODUCTION

Aye (2011) is a data mining technique that derives the interesting data from the World Wide Web. More (Litecky *et al.*, 2010), the web content mining is a part of web mining that aims on the raw data exists in the web pages; the source data primarily contains textual data in web pages (e.g., words, but also tags), general applications are content-based categorization and content-based ranking of web pages. Web structure mining is also a section of web mining that aims on the structure of web sites; source data mainly contains the structural data in web pages (e.g., links to other pages); general applications are link based classification of web pages, ranking of web pages through the mixture of content and (Kumar and Singh, 2010) and reverse engineering of web site models. Web usage mining is a section of web mining that derives the knowledge from server log files, source data primarily

contains (textual) logs that are gathered when the users access the web servers and may be represented in standard formats; general applications are those derived from user modelling technique such as web personalization, adaptive web sites and user modeling (Robal and Kalja, 2012).

An action that adjusts the data or services provided by a web site to the requirements of a specific user or a set of users taking advantage of the knowledge procured from the users' navigational behavior and individual interests in assortment with the content and the structure of the web (Pierrakos and Paliouras, 2010) is called web personalization. The motto of web personalization technique is to provide users with the data they required, without demanding them to ask for it (Papagelis and Zaroliagis, 2012). From an architectural and algorithmic point of view, personalization systems divided into three categories: Rule based systems, content filtering systems and collaborative filtering (Liu *et al.*, 2012). In rule

Corresponding Author: Prince Mary, S., Department of Computer Science Engineering, Sathyabama University, Chennai-119, India

based filtering system, the users are asked to answer a set of questions and these questions are extracted from a decision tree. When the users proceed to answer it, they eventually receive a result (e.g., a list of products) which is tailored to their requirements. Content based filtering system is based on the individual user's preference. The system notices the attitudes of every user and suggests items to the users based on the similar items liked by different users in the past. Collaborative filtering systems asks users to rate the objects or reveal their preferences and interests and then return the data that is predicted to be of interest to them. This is based on the assumption that the users with the same attitude (e.g., users that rate similar objects) have analogous interests. Content based filtering system, rule based filtering system and collaborative based filtering system are together used for inferring more accurate.

In this study we have recommended a web page recommendation based on genetic algorithm. Our process contains three phases as data preparation, mining of informative rules and recommendation. The data preparation is the process of data preprocessing and the identification of users. The mining of informative rules is the phase that uses genetic algorithm. With the new fitness function, genetic algorithm is used to mine the informative rules from the database with the consideration of diverse constraints such as time of stay, quality and recent visiting. The process involved in genetic algorithm is fitness calculation from the initial population, crossover and mutation. The web page sequences with best fitness and which satisfies the constraints are go to the next process which is crossover. After crossover, the web sequences which have best fitness values are take for the mutation process. The web sequences after mutation process is stored separately. The process is repeated from the beginning using the web page sequences we obtained after mutation process and the web sequences we obtained after crossover process with best fitness values and the web sequences we obtained with best fitness and that satisfies all the constraints after calculating fitness of web sequences from initial population. The process would be repeated until we get the best solution after mutation process. The best solution would be the web page sequences of m number of users. The last phase is to form a recommendation tree using the best solution we obtained after mutation process.

A lot of web page recommendation algorithms in terms of web usage mining and soft computing were presented in literature. We brief some of the latest works presented in the literature for web page recommendation. A technique that derives more informative rules from click streams for web page recommendation based on genetic

programming and association rules were recommended by Jaekwang *et al.* (2012). Initially, they split the click streams into sub-click streams using the contexts for generating more informative rules. To split the click streams in consideration of context, they have extracted six features from users' navigation logs. A set of split rules were formed by merging those features using genetic programming and the informative rules for recommendation were derived using association rule mining algorithm.

An E-commerce website recommendation using the features of users and goods similarities distribution has been suggested by Dian (2010). They have developed an E-commerce recommendation technique derived from clustering using genetic algorithm. Using a composite weight matrix, the situation of users purchasing was integrated and this technique was accorded with the reality of E-commerce website personal service and has been perfected for users and goods clustering computing on E-commerce website recommendation and also the technique improved the result of clustering. The accuracy of similarities of users and goods depends on the result of the clustering.

A genetic algorithm based automatic web page classification system has been suggested by Ozel (2011). Their system uses both HTML tags and the terms belong to each tag as classification feature and learn the optimal classifier from the positive and negative web pages in the training dataset. However, both of them were taken together and the optimal weights for the features were learned by genetic algorithm. The accuracy of the classification is improved by using the HTML tags and the terms in each tag separately and the number of documents in the training dataset affects the accuracy if the number of negative documents was larger than the number of positive documents in the training dataset.

A recommender system for music data was proposed by Kim (2010). Their system combines two techniques which are content-based filtering technique and interactive genetic algorithm. The motto of their system is to effectively adapt and respond to the instant changes in users preferences. Their experiment was done in an objective manner and displays that their system can recommend items suitable with the subjective favorite of each user. Kim and Ahn (2012) have suggested a recommender system that effectively adjusts and instantly respond to any alterations in the system by using the content based filtering within the framework of interactive evolutionary computation. Besides, a data grouping model was employed to increase the computational time efficiency. Their numerical result showed that their system provided more accurate music suggestions than other content based filtering systems.

An effective and scalable model to settle the web page recommendation issue has been recommended by Forsati *et al.* (2009). To learn the behavior of the previous users, they have used the distributed learning automata and cluster pages based on the learned pattern. One of the challenging issues in recommendation systems was managing the unvisited or newly added pages. They need to provide an opportunity for these rarely visited or newly added pages to be integrated in the recommendation set, as they never were recommended. Considering that issue and introducing a weighted association rule mining algorithm, they presented an algorithm for recommendation. To extend the recommendation set, they have employed the HITS algorithm. They evaluated their proposed algorithm under diverse settings and showed how the technique improved the overall quality of web recommendations.

Web page recommendation based on weighted association rules has been proposed by Forsati and Meybodi (2010). To solve the web page recommendation issue, they have presented three algorithms. In their first algorithm, the distributed learning automata were used to learn the behavior of previous users and using the learned patterns, it would recommend the pages to the current user. In their second algorithm, they used weighted association rule mining algorithm for recommendation. In their third algorithm, they combined both the algorithms to improve the efficiency of web page recommendation.

2. MATERIALS AND METHODS

Our proposed method has three phases as data preparation, mining of informative rules and recommendation.

2.1. Data Preparation

The data preparation contains the process such as data preprocessing and the user identification to process the dataset in our technique. The preprocessing of data and the user identification is explained below.

2.2. Data Preprocessing

The web log file contains the following details as follows: IP address, access time, HTTP request type used, URL of the referring page and browser name. An example for web server log file is as follows: 192.174.54.23 [28/Nov/2012:10:55:38] "GET/HTTP/1.1" http://www.loganalyzer.net/ "Mozilla/17.0 Windows 07.

2.3. User Identification

The user identification is an essential step to form the sequential database. To classify the users with same IP

address, we need to consider the session. Hence, the IP address and the user session are used to track a unique user from the web log file. The unique IP address is a new user but simultaneously we have to set a particular time period to classify different users with same IP address. When the time period which we set is reached, then from the next second it is consider as a new user for the same IP address.

2.4. Mining of Informative Rules

This is the second phase of our technique that mines the informative rules using genetic algorithm. The **Fig. 1** shows the block diagram of our proposed web page recommendation using genetic algorithm.

The **Fig. 1** delineates as follows: We are choosing some chromosomes from the database and calculating the fitness values for each chromosome which we chosen from the database. The chromosomes are nothing but the sequence of web pages visited by different users and formed as a tree. After finding the fitness values for each chromosome that we chosen from the database, we have to select *m* chromosomes which have best fitness values. Thereafter, we have to find the crossover using the chromosomes which we obtained with best fitness values. After applying the crossover on the chromosomes which we obtained with, best fitness values, we need to calculate the fitness. Using the fitness values after applying the crossover, we have to select *m* chromosomes with best fitness. Thereafter, we need to apply the mutation process on the chromosomes with best fitness values we obtained after applying the crossover. Then, we need to store the *m* chromosomes we procured after mutation process. The *m* best fitness chromosomes we obtained from the initial chromosomes, crossover chromosomes and mutation chromosomes are together taken as the initial chromosomes and the process will go on until we get the best solution. After we get the best solution, the *m* fitness chromosomes which we stored eventually are used to form the recommendation tree.

2.5. Initial Population

2.5.1. Definition

It is the dataset of different users that are taken from the database randomly.

The initial population is the dataset of different users. The dataset contains the list of web pages that are arranged in sequence. The sequence depends on the web pages visited by that particular user. The sequences of web pages are then formed as trees randomly. The **Fig. 2** shows the sample web pages visited by different users formed as trees.

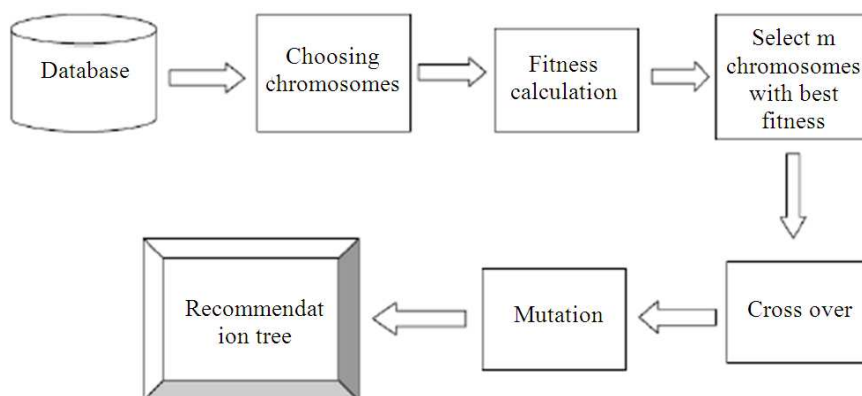


Fig. 1. Block Diagram of our proposed technique

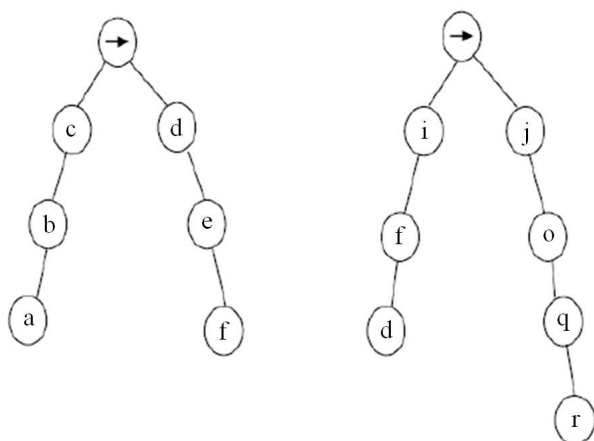


Fig. 2. Sample web page sequences visited by different users

2.6. Fitness Calculation

2.6.1. Definition

It is the calculation to find the fitness of each tree to allow it for the next process based on the fitness value.

After getting the initial population from the database, we have to find the fitness values f for the datasets of different users which we used as initial population. The fitness value is the sum of three factors as first factor, second factor and the third factor:

$$f = f_1 + f_2 + f_3$$

Where:

f = Fitness

f_1 = First factor

f_2 = Second factor

f_3 = Third factor

The first factor f_1 is the ratio of the frequency of dataset (LN+ RN) of a particular user to the total datasets TD in the database. It is denoted as equation below:

$$f_1 = \frac{f(LN + RN)}{TD}$$

Where:

LN = Left said nodes

RN = Right said nodes

TD = Total data set in the database

The second factor f_2 is the ratio of the frequency of dataset (LN+ RN) of a particular user to the frequency of sequence of web pages in the left side nodes LN of the same user. It is shown as equation below:

$$f_2 = \frac{f(LN + RN)}{f(LN)}$$

The third factor f_3 is the ratio of the frequency of sequence of web pages in the left side nodes LN of a particular user to the frequency of sum of sequence of web pages in the left side nodes and the first web page of the right side nodes of the same user. It is shown as equation below:

$$f_3 = \frac{f(LN)}{f(LN + \text{first element of RN})}$$

The Fig. 3 shows the sequence of web pages in the left side nodes and the sequence of right side nodes and first element of the right side nodes.

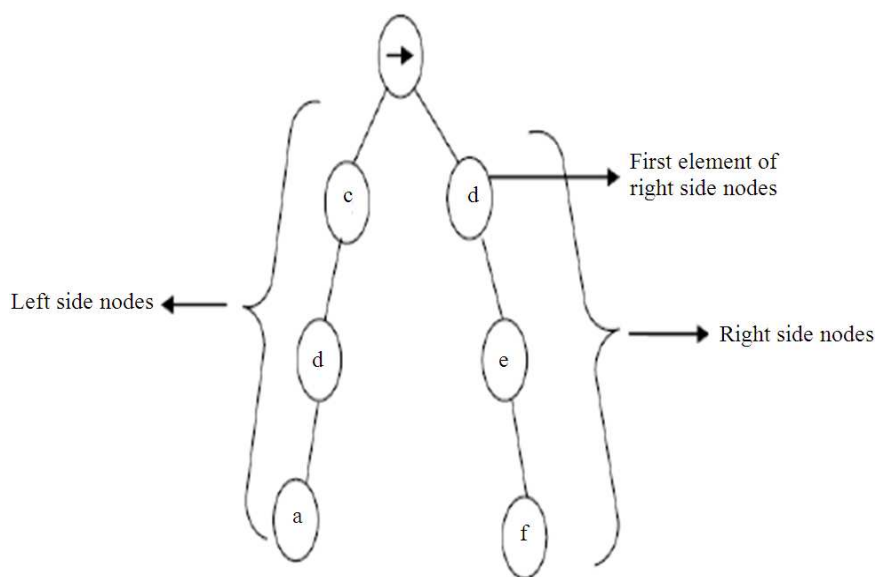


Fig. 3. Sequence of web pages in LN and RN

After finding the fitness values, we have to check the time duration N_{TD} , quality N_Q and recent visit N_{RV} with the threshold values of the time duration quality $Q_{threshold}$ and recent visit $RV_{threshold}$. If the values are greater than the threshold values we set, the trees which have the values greater than the threshold values are then taken for the crossover. The process of checking the time duration, quality and recent visit of a particular dataset is as follows.

First we have to sum the time duration of all the web pages in a particular dataset of a user and we have to check the solution of that calculation with the threshold value of the time duration which we set. The calculation is explained by equation below:

$$N_{TD} = \sum_{i=1}^L N_i^{td}$$

Where:

N_{TD} = Total time duration of nodes in a tree

N_i^{td} = Each node in a tree

L = Total Number of nodes in a tree

Thereafter, we have to check whether the total time duration of the nodes in a tree N_{TD} is greater than the threshold time duration $TD_{threshold}$ which we set. Then, we need to add the quality rate of all the web pages in a dataset and check the solution with the quality threshold which we set. The calculation of total quality of a dataset is shown as equation below:

$$N_Q = \sum_{i=1}^L N_i^q$$

where, $N_Q \rightarrow$ Total quality of nodes in a tree.

After calculating the total quality N_Q of a tree, we have to check it with the threshold quality $Q_{threshold}$ we set whether the total quality N_Q of the tree is greater than the threshold quality $Q_{threshold}$. Thereafter, we have to check the last node N_i of the tree with the threshold value $RV_{threshold}$ whether it is greater than the threshold value $RV_{threshold}$. If the total time duration N_{TD} , total quality N_Q and recent visit N_{RV} of a particular tree are greater than the threshold values we set, then that particular tree:

$$EC = \begin{cases} \text{Aallwo, if } N_{TD} > TD_{threshold} \text{ and} \\ N_Q > Q_{threshold} \text{ and } N_{RV} > RV_{threshold} \\ \text{deny,} & \text{else} \end{cases}$$

where, $EC \rightarrow$ Eligible for crossover process.

2.7. Crossover

2.7.1. Definition

It is the process of interchanging particular sequence of nodes of two trees.

The trees which have best fitness values and that satisfy the threshold conditions would come to the crossover process. The crossover process is that interchanging certain

sequence of nodes between two trees. The **Fig. 4** shows two sample parent trees that have satisfied the fitness value and the threshold values which we set and the **Fig. 5** shows the trees we obtained after the crossover process.

The m trees we obtained with best fitness and satisfy the threshold conditions are used to find the crossover by comparing the trees with each other. The **Fig. 6** shows a sample block diagram of crossover process.

This block diagram shows that the m trees which we obtained with best fitness that satisfies the threshold values are grouped as two parent trees to form two new trees. The two new trees are formed by fix a point on both the parent trees which we considered and by interchange the sequence of nodes after that point in both the parent trees. Similarly, we have to find the crossover for all the trees by comparing with each other. The fixed point would be same for all the m trees. After finding the crossover for all the trees by comparing with each other, we need to find the fitness values for all the newly formed trees.

Thereafter, we have to select m trees that have best fitness values. The selected m trees after the chromosome process using the fitness values are then subjected to mutation process.

2.8. Mutation

2.8.1. Definition

It is the process of altering a particular node as different node.

The m trees that has best fitness values after the chromosome process is subjected to the mutation process.

The mutation process is as follows: it would choose a particular node to alter and it would change that chosen node as a different node i.e., the web page in that chosen node will get change. The chosen node would be same for all the m trees but perhaps with different web page. The **Fig. 7** shows the point fixed for the mutation process and the **Fig. 8** shows a sample tree after the mutation process.

After the mutation process, we have to store the trees SD_i we obtained and we need to do the process from the beginning i.e., instead of taking the initial population we have to take the m trees we obtained by means of best fitness before the crossover and after the crossover process and after the mutation process. Thereafter, we need to do the same process from the beginning except the threshold checking. The iteration will go on until we get the constant trees SD_i .

2.9. Recommendation Tree

2.9.1. Definition

It is the technique of structuring a tree from the final trees we obtained after mutation process.

The recommendation tree is formed by combining the m trees we obtained eventually after the mutation process. The **Fig. 9** shows the sample trees we obtained eventually after the mutation process and the **Fig. 10** shows the sample recommendation tree formed using the final trees after mutation.

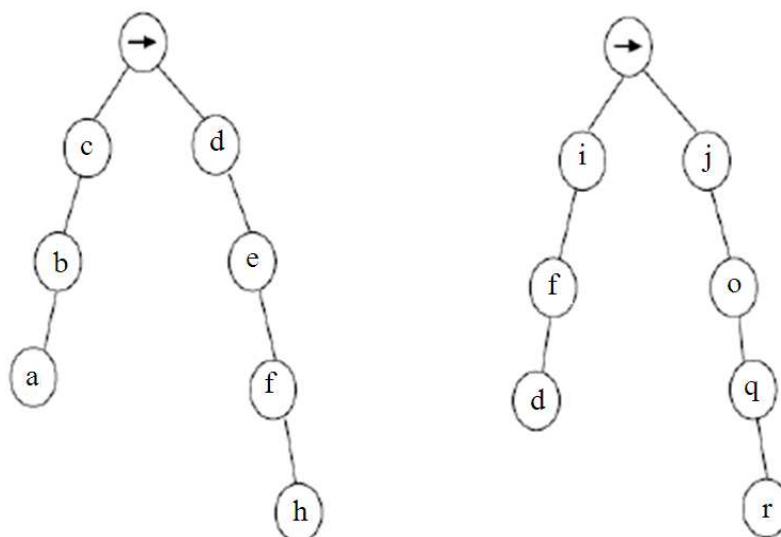


Fig. 4. Sample parent trees used for crossover process

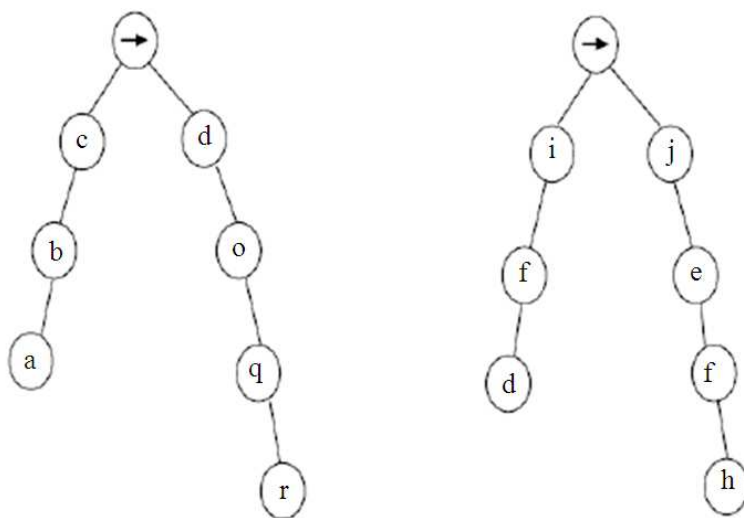


Fig. 5. Trees obtained after crossover

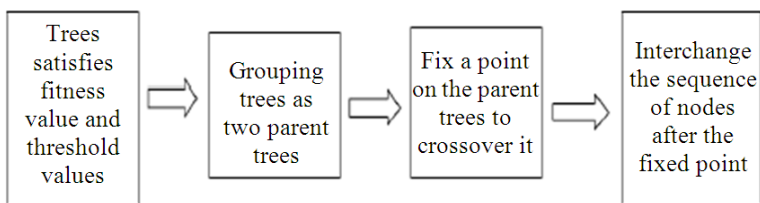


Fig. 6. Sample block diagram for the process of crossover fitness

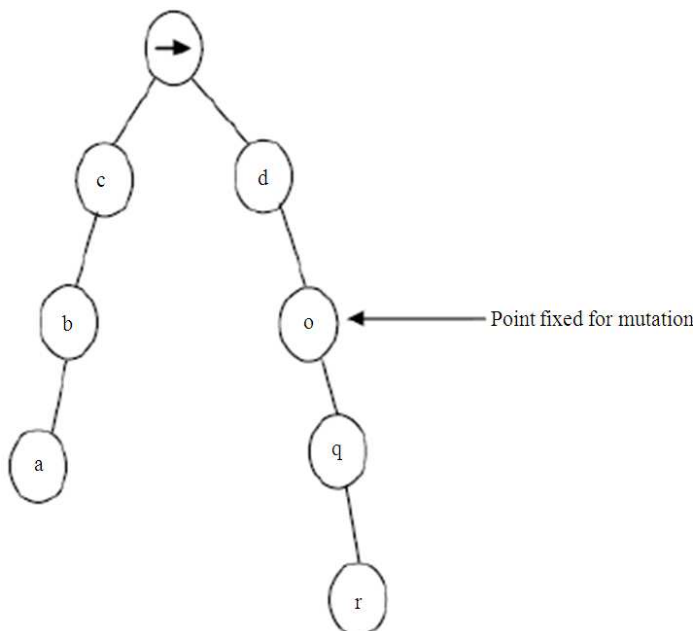


Fig. 7. Sample tree with point fixed for mutation

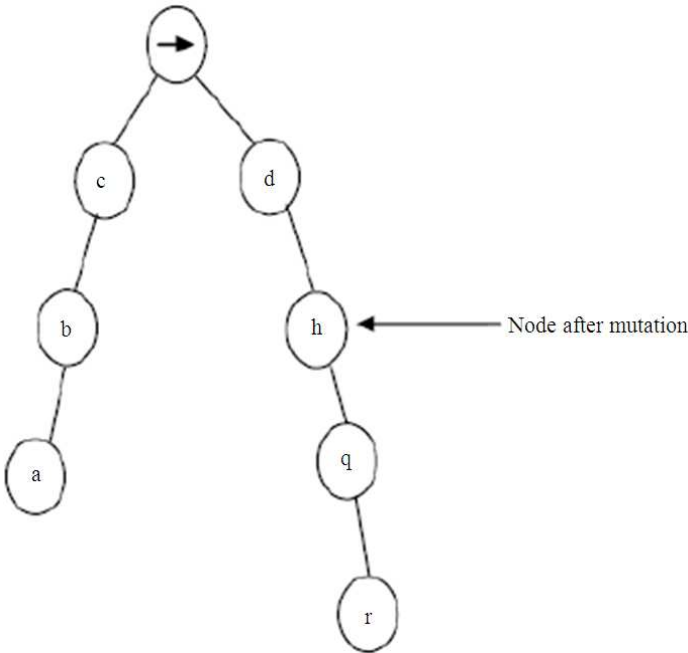


Fig. 8. Sample trees obtained after mutation

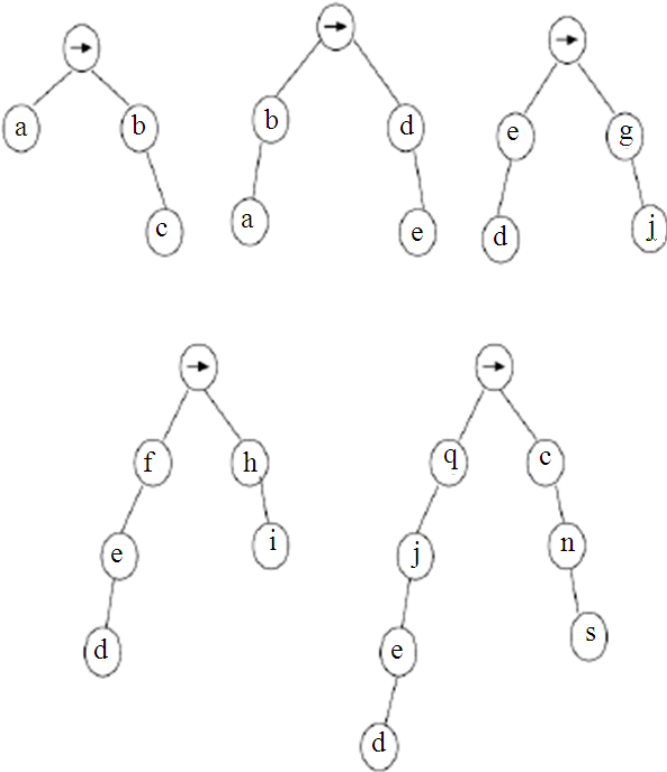


Fig. 9. Sample recommendation tree

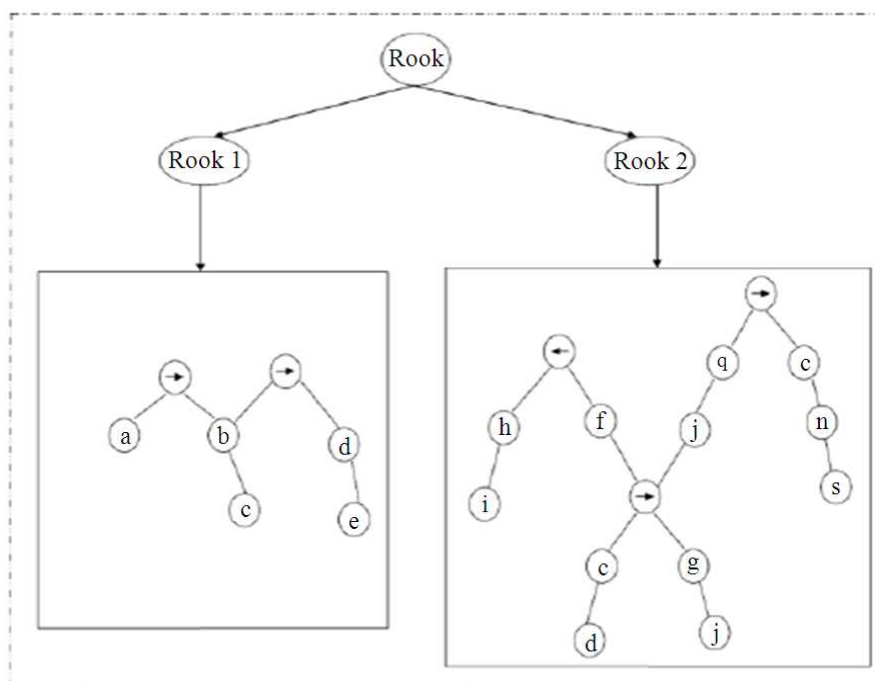


Fig. 10. Sample recommendation tree

The Fig. 9 explains as follows: The first tree denotes the sequence of web pages as abc, the second tree denotes the sequence of web pages as abde, the third tree denotes the sequence of web pages as deg j, the fourth tree denotes the sequence of web pages as defhl and the fifth tree denotes the sequence of web pages as abcabdej.

The Fig. 10 explains as follows: the tree in the root1 is formed using the web sequences abc and abde. The tree in the root2 is formed using the web sequences deg j, defhl and dejqns. If we link the degj in the tree in root1, then it would be formed as abdeg j which is wrong. So we have formed a separate root to form the recommendation tree for the web sequences starts with de.

3. RESULTS AND DISCUSSION

We have used two datasets for our recommended technique which are synthetic dataset and the dataset by crawling del.icio.us site and the results are evaluated with precision, applicability and hit ratio.

3.1. Experimental Setup and Dataset Description

Our proposed technique is applied in java (jdk 1.6) with the system that has i5 processor and 4GB RAM.

The synthetic dataset is generated as the same format of the real datasets and the dataset from the del.icio.us site is formulated as follows: The user id is taken as user, the bookmark id is taken as web page, the tag id is taken as time duration and the quality and recent visit are generated randomly. The performance of our technique is evaluated using some evaluation metrics.

3.2. Evaluation Metrics

Our proposed technique is evaluated using precision calculation, applicability calculation and hit ratio Niranjana *et al.* (2010a; 2010b). The calculation of the precision, applicability and hit ratio are as follows.

The precision calculation is defined as number of correct recommendations which is divided by the sum of number of correct recommendations and the number of incorrect recommendations:

$$\text{Precision} = \frac{\text{no.of correct recommendation}}{\text{no.of correct recommendation} + \text{no.of incorrect recommendation}}$$

The applicability calculation defined as the sum of number of correct recommendations and number of incorrect recommendations which is divided by the total number of given requests:

$$\text{Applicability} = \frac{\text{no.of correct recommendation} + \text{no.of incorrect recommendation}}{\text{total no.of given request}}$$

The precision calculation is defined as it is the product of precision value and applicability value:

$$\begin{aligned} \text{Hit ratio} &= \text{precision} \times \text{Applicability} \\ &= \frac{\text{no.of correct recommendation}}{\text{total no.of given request}} \end{aligned}$$

3.3. Performance Comparison

We have calculated the precision, applicability and hit ratio for synthetic dataset and the dataset from del.icio.us site. The performance using these two datasets is explained as follows.

3.4. Synthetic Dataset

The evaluation metrics are calculated for synthetic dataset using two different threshold values. The first threshold value is mentioned as T1 that has the time duration as 300, quality as 3 and recent visit as 1 and the second threshold is mentioned as T2 that has the time duration as 200, quality as 2 and recent visit as 1.

The **Fig. 11** shows the precision values using different initial population for two different threshold values. When the initial population is 800, the precision value we obtained for the first threshold is 0.9859 and the precision value for second threshold is 0.9 and when the initial population is 700, the precision value is 0.9624 for the first threshold and 0.89 for the second threshold. When the initial population is 600, the precision value is 0.9154 for the first threshold and 0.891 for the second threshold. When the initial population is 500, the precision value is 0.9132 for the first threshold and 0.88 for the second threshold.

The **Fig. 12** explains the applicability values for different initial populations with two different threshold values using synthetic dataset. Here, the applicability value is one for the first threshold when we varying initial population as 800, 700, 600 and 500. The applicability value for second threshold is 0.9 when the initial population is 800, 700 and 600 and when the initial population is 500; the applicability value is 0.8 for the second threshold.

The **Fig. 13** explains the hit ratio for the synthetic dataset for two different threshold values. When the initial population is 800, the hit ratio is 0.9859 for the first threshold and it is 0.9 for the second threshold. When the initial population is 700, the hit ratio is 0.9624 for the first

threshold and it is 0.89 for the second threshold. When the initial population is 600, the hit ratio is 0.9154 for the first threshold and 0.891 for the second threshold. When the initial population is 500, the hit ratio is 0.9132 for the first threshold and 0.879 for the second threshold.

3.5. Dataset from del.icio.us

The evaluation metrics are calculated for this dataset using two different threshold values. The first threshold is mentioned as T1 and the second threshold is mentioned as T2. The threshold T1 has the time duration as 300, quality as 3 and the recent visit as one. The threshold T2 has the time duration as 200, quality as 2 and the recent visit as one.

The **Fig. 14** explains the precision values for the dataset we taken from del.icio.us site for two different thresholds. The precision values we obtained for the first threshold are 0.9632, 0.9245, 0.9352 and 0.9144 for the respective initial populations 800, 700, 600 and 500. The precision values we obtained for the second threshold are 0.88, 0.845, 0.84 and 0.82 for the respective initial populations 800, 700, 600 and 500.

The **Fig. 15** explains the applicability values for the dataset we taken from del.icio.us site for two different thresholds T1 and T2. The applicability value is 1 using the first threshold for all the initial population we set and the applicability value is 0.82 using the second threshold for all the initial population we set.

The **Fig. 16** shows the hit ratio values for the dataset taken from the del.icio.us for two different thresholds. The hit ratio values for the first threshold are 0.9632, 0.9245, 0.9352 and 0.9144 for the respective initial populations 800, 700, 600 and 500. The hit ratio values for the second threshold are 0.88, 0.845, 0.84 and 0.82 for the respective initial populations 800, 700, 600 and 500.

The **Table 1** shows the values of precision, applicability and hit ratio we obtained for our technique and the prefix span algorithm using the synthetic dataset. Here, the precision value, applicability value and the hit ratio values of our technique is the average values using the first threshold and the second threshold and the respective average values for the prefix span algorithm. In our technique, we took the precision values, applicability values and hit ratio values for different initial populations and in prefix span algorithm, the precision values, applicability values and hit ratio values are taken for different support values. Here, the average percentage of our technique shows better performance than the prefix span algorithm.

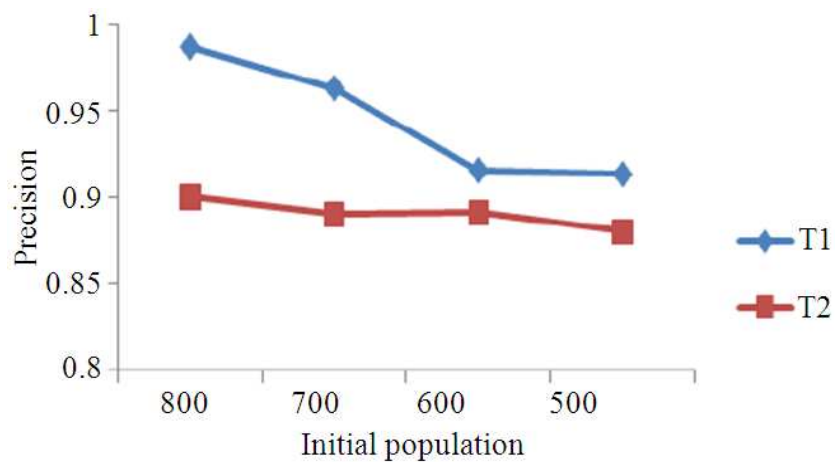


Fig. 11. Precision for synthetic dataset

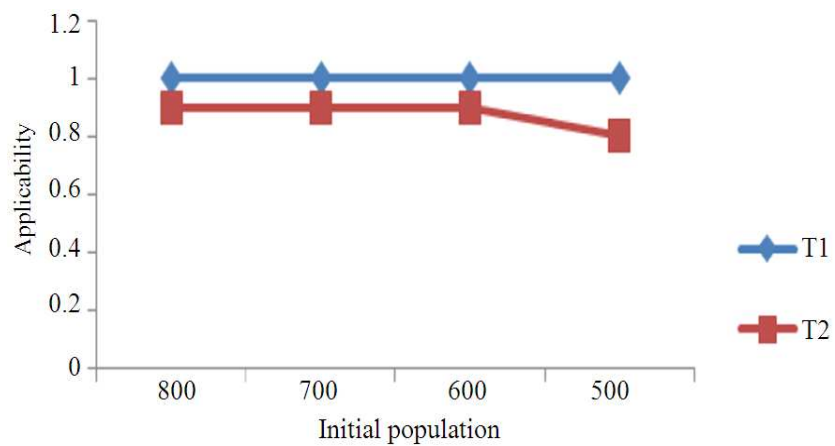


Fig. 12. Applicability for synthetic dataset

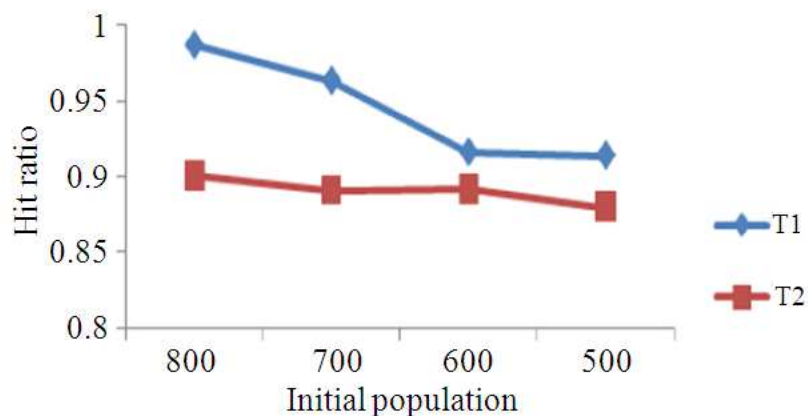


Fig. 13. Hit ratio for synthetic dataset

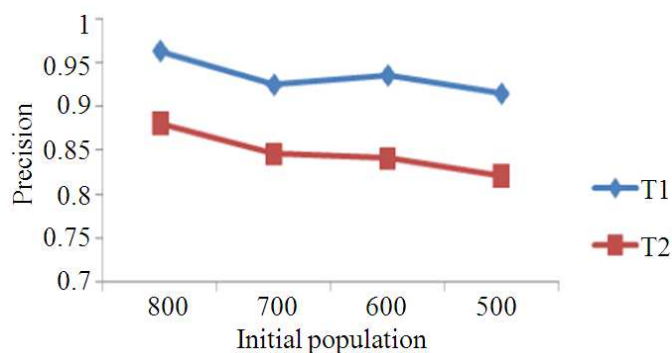


Fig. 14. Precision for the dataset from del.icio.us

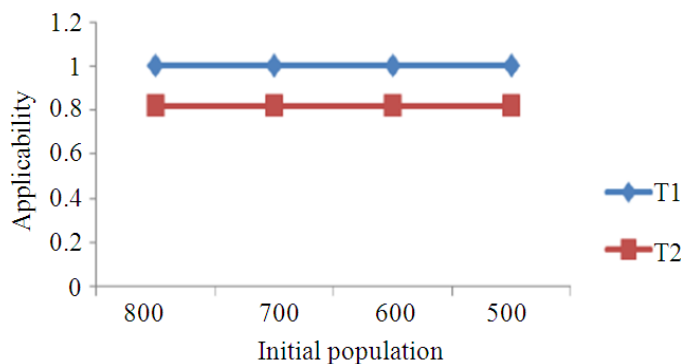


Fig. 15. Applicability for the dataset from del.icio.us

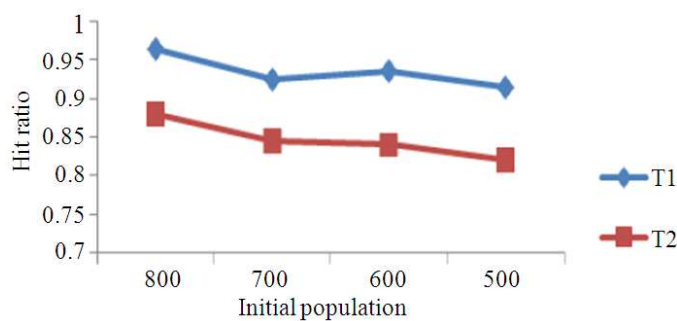


Fig. 16. Hit Ratio for the dataset from del.icio.us

Table 1. Comparison of our technique and prefix span algorithm

	Our technique		Prefix span
	T1	T2	
Precision	94.4225	89.025	45.6093
Applicability	100.0000	87.500	98.8732
Hit ratio	94.4225	89.000	45.0704

4. CONCLUSION

In this study we have proposed a technique for web page recommendation using genetic algorithm. Our proposed technique contains three phases as data preparation, mining of informative rules and recommendation. The sequence of web pages visited by different users was identified using the first phase. The

identified data from the first phase is used for mining in the second phase by means of genetic algorithm. The sequences of web pages with best solution were identified by the genetic algorithm using its three stage process which is fitness value calculation, crossover and mutation. We have set three threshold values to allow the process for next level after the initial fitness calculation. The threshold value is checked at the first time of the fitness calculation alone. The three stage process of the genetic algorithm is repeated until we obtained the best solutions. The recommendation tree is formed using the best solutions we obtained using genetic algorithm. The performance of our technique is calculated for two different datasets with two different threshold values. Our technique is compared with the prefix span algorithm and our technique shown better average percentage than the prefix span algorithm.

5. REFERENCES

- Aye, T.T., 2011. Web log cleaning for mining of web usage patterns. Proceedings of the 3rd International Conference on Computer Research and Development, Mar. 11-13, IEEE Xplore Press, Shanghai, pp: 490-494. DOI: 10.1109/ICCRD.2011.5764181
- Dian, H., 2010. E-commerce recommendation method based on genetic algorithm and composite weight matrix. Proceedings of the Electrical and Control Engineering International Conference, Jun. 25-27, Wuhan, pp: 2760-2763. DOI: 10.1109/iCECE.2010.674
- Forsati, R. and M.R. Meybodi, 2010. Effective page recommendation algorithms based on distributed learning automata and weighted association rules. Expert Syst. Applic., 37: 1316-1330. DOI: 10.1016/j.eswa.2009.06.010
- Forsati, R., M.R. Meybodi and A. Rahbar, 2009. An efficient algorithm for web recommendation systems. Proceeding of the International Conference on Computer Systems and Applications, May 10-13, IEEE Xplore Press, Rabat, pp: 579-586. DOI: 10.1109/AICCSA.2009.5069385
- Jaekwang, K.I.M., K.H. Yoon and J.H. Lee, 2012. An approach to extract informative rules for web page recommendation by genetic programming. IEICE Trans. Commun., 95: 1558-1565.
- Kim, H.T. and C.W. Ahn, 2012. An interactive evolutionary approach to designing novel recommender systems. Int. J. Phy. Sci., 15: 2327-2338.
- Kim, H.T., 2010. A recommender system based on genetic algorithm for music data. Proceedings of the 2nd International Conference Computer Engineering and Technology, Apr. 16-18, IEEE Xplore Press, Chengdu, pp: V6-414-V6-417. DOI: 10.1109/ICCET.2010.5486161
- Kumar, P.R. and A.K. Singh, 2010. Web structure mining: Exploring hyperlinks and algorithms for information retrieval. Am. J. Applied Sci., 7: 840-845. DOI : 10.3844/ajassp.2010.840.845
- Litecky, C., A. Aken, A. Ahmad and H.J. Nelson, 2010. Mining for computing jobs. IEEE Soft., 27: 78-85. DOI: 10.1109/MS.2009.150
- Liu, Q., E. Chen, H. Xiong, C.H.Q. Ding, and J. Chen, 2012. Enhancing collaborative filtering by user interest expansion via personalized ranking. IEEE Trans. Syst. Man Cybernet., 42: 218-233. DOI: 10.1109/TSMCB.2011.2163711
- Niranjan, U.R., B.V. Subramanyam and V. Khana, 2010b. An efficient system based on closed sequential patterns for web recommendations. Int. J. Compu. Sci., 7: 26-34.
- Niranjan, U.R., B.V. Subramanyam and V. Khanaa, 2010a. Developing a web recommendation system based on closed sequential patterns. Commun. Comput. Inform. Sci., 101: 171-179. DOI: 10.1007/978-3-642-15766-0-25
- Ozel, S.A., 2011. A web page classification system based on a genetic algorithm using tagged-terms as features. Expert Syst. Appli., 38: 3407-3415. DOI: 10.1016/j.eswa.2010.08.126
- Papagelis, A. and C. Zaroliagis, 2012. A collaborative decentralized approach to web search. IEEE Trans. Syst. Man Cybernet., 42: 1271-1290. DOI: 10.1109/TSMCA.2012.2187887
- Pierrakos, D. and G. Paliouras, 2010. Personalizing web directories with the aid of web usage data. IEEE Trans. Knowl. Data Eng., 22: 1331-1344. DOI: 10.1109/TKDE.2009.173
- Robal, T and A. Kalja, 2012. Learning from users for a better and personalized web experience. Proceedings of the Technology Management for Emerging Technologies, Jul. 29-Aug. 2, IEEE Xplore Press, Vancouver, BC., pp: 2179-2188.