International Journal of Physical Sciences Vol. 7(10), pp. 1613 - 1621, 2 March, 2012
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.1709
ISSN 1992 - 1950 ©2012 Academic Journals

Full Length Research Paper

A learning automata approach to adaptive web sites

Fatemeh Zandevakili[1]*, Javad Akbari Torkestani[2] and Tayebeh Yarahmadi[1]

[1]Department of Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran.
[2]Young Researchers Club, Arak Branch, Islamic Azad University, Arak, Iran.

Accepted 19 January, 2012

Web is a large network of related documents that are increasing rapidly. Due to the unknown optimal structure for a web site, providing a way to do this is important. In this paper, a method based on distributed learning automaton (DLA) with variable number of actions to create an adaptive web site is provided that has ability to insert efficient links and eliminate inefficient links. In the proposed method, we used web usage data that had tried to change the structure of the web site based on users' interaction in the past. Simulation results show that in the new structure, users can achieve their desired information with less number of steps than primary web site and the proposed method because considering the variable learning parameter has a higher efficiency than existing algorithms and does not need to readjust the learning parameters if numbers of web pages increase.

Key words: Web usage mining, adaptive web sites, site improvement, site reorganization, learning automata.

INTRODUCTION

Extracting information from web documents using data mining techniques is called web mining. In Tyagi et al. (2010), Kosala and Blockeel (2000), Chitraa and Davamani (2010), Tao and Hong (2008), Shahabi and Kashani (2003), Suneetha and Krishnamoorthi (2009) and Thakare and Gawali (2010), web mining is expressed at three levels: web content mining, web structure mining and web usage mining. In the content level, the purpose is exploring web content. In the structure level, the purpose is using the hyperlink topology to determine the relationship between web documents. In the web usage level, the purpose is the discovery of useful information from user's interaction when using the web. One application of web usage mining is improvement of the structure of web sites or creation of adaptive web sites. The purpose of this operation is shortening of users' navigation path to reach the target pages, increasing users' satisfaction and using of that web site.

Adaptation is an approach that focuses on the bases of a web site design such as the ease use and navigation. Today, many web sites have incomprehensible content and confusing navigation. A flexible and simple use of web site navigation gives the users a sense of satisfaction and adds more value to their browsing experience. The term adaptive web sites was introduced by Etzioni and Perkowitz, and defined web sites as those that are capable of changing their structure and presentation based on users' access patterns. Thus indirectly, it can be regarded as an act of modelling the web site in contrast to modelling the users (Mobasher et al., 1999). Adapting web site's structure involves altering the way information is organized within the web site. The underlying idea is to identify items that are frequently accessed by users and making them easier to be found. This allows users to save time and effort in trying to find a piece of information that may be buried deep within a web site's structure. Presentation adaptation involves altering the way information is presented in the web site. All too often users browsing behaviours change rapidly while the web site remains static. The main idea here is to monitor how information in a web site is being viewed by the users and make change to the way this information is presented to them based on the observation (Perkowitz and Etzioni, 2000a). This study focuses on the adapting web site's structure. The difference of adapting and personalizing the web site is that, in adapting the web site, structure and presentation

*Corresponding author. E-mail: zandevakili_msc@yahoo.com.

of a web site based on user preferences are set for all users. An example of these systems is My Yahoo. This process can be done either manually or automatically. But personalization focuses on the adaptation of a group or individual and these changes can be done dynamically.

Baradaran and Meybodi (2007b) presented an algorithm based on learning automata (LA) to make a web site adaptive. So, after observation of behaviour, some users can change structure of the site by using the status of LA. Computer simulations show that the proposed algorithm can reduce users browsing time.

In Anari and Meybodi (2007), a statistical method was presented. In this way, a simple formula is used to obtain the relationship between web pages based on users' navigation path in the web site. The relationship between pages i and page j is defined as the ratio number of times that users have come from page i to page j [N(i, j)], to the number of times that users have come from page i to each other page at the site. The efficiency of this method is less than method presented in Baradaran and Meybodi (2007b).

In Perkowitz and Etzioni (1997b, 1999, 2000a), web usage data is clustering. Using that, the dynamic list of the web pages has been created that with changing, behaviours and interests of users adapts itself. This method is implemented in a web site and its efficiency is shown with measurement of the number of users that used the dynamic list.

Perkowitz and Etzioni (2000b) studied about adaptation of web sites. In this paper, a way is presented called Index Finder, that is, the only form of study to date that has attempted to adapt a web site's presentation non-destructively based on users browsing patterns. The study investigates the problem of automatically synthesizing new index documents containing a collection of links to documents that are topically related but currently unlinked in the web site. Index Finder is a conceptual cluster mining algorithm that relies on visit coherence assumption to identify clusters of documents that share a common topic and presents them as content (that is, collection of hyperlinks) for candidate index documents.

Lee and Shiu (2004) introduced a metric called web site's efficiency, and it proposed an algorithm that improves structure of the web site based on users' access patterns by adding and deleting hyperlinks in the web site. In this study, the authors claim that the longer a user's traversal path is, the higher probably are for him to make mistakes. Method introduced in this article allows users to access the target document by clicking fewer and at the same time, improves the movements of web site by identifying and eliminating inefficient links. The weakness of this study is that researchers do not clearly describe how two documents in the same path are chosen to create a shortcut.

Shokry et al. (2006) presented a new method using fuzzy clustering for creation dynamic list pages. Simulation results using this method are shown more efficient than method presented in Perkowitz and Etzioni (2000a).

In Weigang et al. (2002), the structure of a web site is changed using ant colonies and to increase the number of users that have reached its goal in a given period of time. In Lin (2006), adaptation of a web site for increased use of users of all links is modelled as an optimization problem.

In Tiwari et al. (2010), the content and links between web pages were introduced as two important factors in web site navigation and is used as the fuzzy relationship between them to optimize the web site architecture.

In Abedin and Sohrabi (2009), using the paths observed by the users, the site is modelled as a directed graph, and then introduce measures based on graph theory to improve the structure of web site. Finally, a strategy also has been proposed to improve ranking.

Perkowitz and Etzioni (1997a) introduced a non-destructive organization adaptation called "promotion and demotion" that makes a hyperlink or a document easier to be found by placing a reference to it in the index document. Brickell et al. (2007) proposed a non-destructive adapting algorithm of web site that makes a shortcut between documents of a web site and promote them on the site index page.

Zhu et al. (2002) proposed a way to create a Markov model of a web site based on previous visitors' behaviour. This Markov model is used to make link predictions that assist new users to navigate the web site. Also, a maximal forward path method is used to further improve the efficiency of link prediction. Finally, link prediction has been implemented in an online system called Online Navigation Explorer (ONE) to assist users' navigation in the adaptive web site.

Method presented in this paper, unlike most existing methods, have investigated all the pages in the users' navigation path using distributed LA (DLA) and tried to provide optimal structure for web sites that has ability to insert efficient links and eliminate inefficient links. The difference of this method with existing methods, is considering the variable learning parameter based on stay time, length of route and topic correlation of two pages (that is, Reverse of Euclidean distance between the content vector of two pages viewed) that extract more accurate information from the user's sessions and so is considering restriction to prevent excessive links on website based on topic correlation of the pages. Also, the proposed algorithm has two advantages than the only reported method based on LA, high efficiency and no need to readjust the learning parameters of LA if numbers of web pages increase. LA is used in such cases: identify similarity between web pages (Baradaran and Meybodi, 2007a), ranking web pages (Saati and Meybodi, 2006) and web personalization (Forsati et al., 2007).

LEARNING AUTOMATA

LA are adaptive decision-making devices operating on unknown random environments. The automata approach to learning involves the determination of an optimal action from a set of allowable actions. An automaton can be regarded as an abstract object which has finite number of possible actions. In each decision process, the automata select an action from its finite set of actions. This action is applied to a random environment. The random environment evaluates the selected action and gives a grade to applied action of automata. The random response of environment (that is, grade of action) is used by automata in further action selection. By continuing this process, the automata learn to select an action with best grade. The learning algorithm used by automata to determine the selection of next action from the response of the environment. An automaton acting on unknown random environment and improves its performance in some specified manner, is referred to as LA. LA can be classified into main categories: fixed structure LA and variable structure LA (Narendra and Thathachar, 1989). The variable structure LA used in this paper is described as follows:

Variable structure LA is represented by quintuple <α ,β , *p*, *T*(α ,β , *p*) > , where $\alpha \equiv \{\alpha_1,\alpha_2,...\alpha_r\}$, $\beta \equiv \{\beta_1,\beta_2,...,\beta_r\}$, and $p \equiv \{p_1, p_2,..., p_r\}$are an action set with *r* actions, an environment response set, and the probability set *p* containing *r* probabilities, each being the probability of performing every action in the current internal automaton state, respectively. The function of *T* is the reinforcement algorithm, which modifies the action probability vector p with respect to the performed action and received response. If the response of the environment takes binary values LA model is P-model and if it takes finite output set with more than two elements that take values in the interval [0, 1], such a model is referred to as Q-model, and when the output of the environment is a continuous variable in the interval [0, 1], it is referred as S-model.

It is evident that the crucial factor affecting the performance of the variable structure LA is learning algorithm for updating the action probabilities. Various learning algorithms have been reported in the literature. Let α be the action chosen at step n as a sample realization from probability distribution p. The linear reward-inaction algorithm is one of the learning schemas and its recurrence equation for updating action probability vector p is defined as Equations 1 and 2.

Reward: $$\begin{aligned} &p_i(n+1) = p_i(n) + a[1 - p_i(n)] \\ &p_j(n+1) = (1-a)p_j(n) \qquad \forall j,\ \ j \neq i \end{aligned} \tag{1}$$

Penalty: $$\begin{aligned} &p_i(n+1) = (1-b)p_i(n) \\ &p_j(n+1) = \frac{b}{r-1} + (1-b)p_j(n) \qquad \forall j,\ \ j \neq i \end{aligned} \tag{2}$$

Where 0<α <1 is called step length and determines the amount of increases (decreases) of the action probabilities.

The mentioned LA has a fixed number of actions. In some applications, like our first proposed algorithm, we need that LA has a changing number of actions (Thathachar and Bhaskar, 1987). A LA with changing number of actions at any time instance n selects its action from a set of active actions *V* (*n*) and behaves like this. For selecting an action, the LA first computes the sum of its actions' probability *K*(*n*) and then the vector *p*ˆ (*n*) is computed according to Equation 3). The automaton selects one of its active actions randomly based on actions probabilities, that is, *p*ˆ (*n*). The automaton applies the selected action *i* α to the environment and gets the response. For desirable responses, the *p*ˆ (*n*) vector is updated based on Equation 4 and undesirable action is updated based on Equation 5. Finally, the automaton updates the actions' probability vector *p*(*n*) based on vector *p*ˆ (*n* +1) as shown in Equation 6.

$$K(n) = \sum_{\alpha_i \in V(n)} p_i(n) \tag{3}$$

$$\hat{p}_i(n) = prob[\alpha(n) = \alpha_i \mid V(n) \text{ is set of active actions}, \alpha_i \in V(n)] = \frac{p_i(n)}{K(n)}$$

$$\begin{aligned} &\hat{p}_i(n+1) = \hat{p}_i(n) + a.(1 - \hat{p}_i(n)) \qquad \alpha(n) = \alpha_i \\ &\hat{p}_j(n+1) = \hat{p}_j(n) - a.\hat{p}_j(n) \qquad \alpha(n) = \alpha_i,\ \forall j\ \ j \neq i \end{aligned} \tag{4}$$

$$\begin{aligned} &\hat{p}_i(n+1) = (1-b).\hat{p}_i(n) \qquad \alpha(n) = \alpha \\ &\hat{p}_j(n+1) = \frac{b}{r-1} + (1-b)\hat{p}_j(n) \qquad \alpha(n) = \alpha_i,\ \forall j\ \ j \neq i \end{aligned} \tag{5}$$

$$\begin{aligned} &p_j(n+1) = p_j(n+1).K(n) \qquad \text{for all j}, \alpha_j \in V(n) \\ &p_j(n+1) = p_j(n) \qquad \text{for all j}, \alpha_j \notin V(n) \end{aligned} \tag{6}$$

Distributed learning automata

A DLA is a network of LA which collectively cooperates to solve a particular problem. The number of actions for a particular LA in DLA is equal to the number of LAs that are connected to this LA. Selection of an action by a LA in the network activates one LA corresponding to the action. Formally, a DLA can be defined by a graph *DLA* = (*A*, *E*), where the set $A = \{A_1, A_2, \ldots, A_n\}$is the set of n LA and $E \subset A \times A$ is the set of edges in the graph. The edge (*i*, *j*) represents the action j of automata LAi. In other words, LAj is activated when action j of automata Lai is

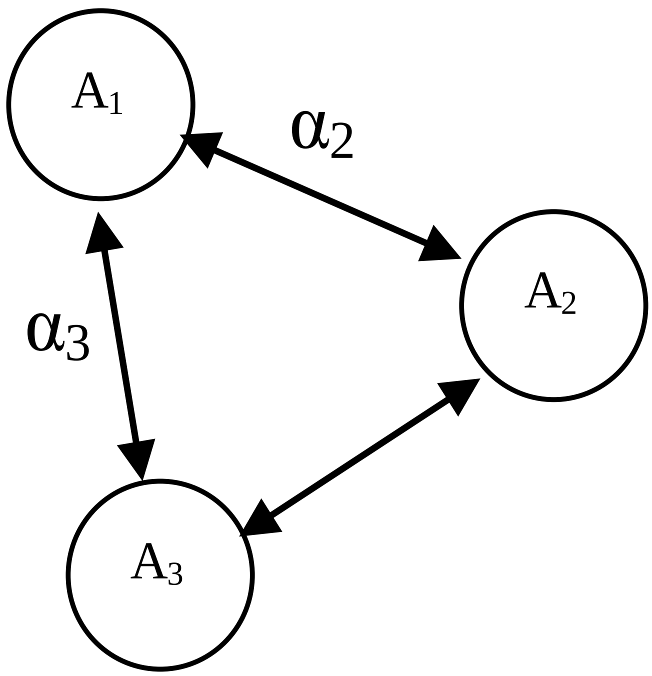

Figure 1. Distributed learning automata.

selected. The number of actions for particular automata LAk (K=1, 2,…, n) is equal to the out-degree of that node. If p j corresponds to the probability distribution of actions of LAj, then pjm shows the probability of selecting action αm by automata Aj. In other words, we can assign a weight to each edge (*i*, *j*) in graph which is equal to the probability of selection of action *i* by automata j (Meybodi and Beigy, 2001; Beigy and Meybodi, 2002, 2006).

For example, in Figure 1, every automata has two actions. Selection of action α3 by A1 will activate automata A3. Activated automata choose one of its actions which results in activation of the LA corresponding to the selected action. At any given time, only one of the automata in the network could be active.

PROPOSED ALGORITHM FOR ADAPTIVE WEB SITE

Due to the unknown optimal structure for a web site, this study is used to reinforcement learning method called DLA (Narendra and Thathachar, 1989) for learning the optimal structure of a web site based on the users' interaction.

In a web site with n pages, we use a DLA with *n* LAs with variable number of actions. For each page in the site, a LA with *n* −1 actions is added to the DLA. Each action corresponds to a page. For each LA at any time, a subset of its actions is active. In the beginning, the number of active actions in the LA assigned to $page_i$ is equal to the number of pages that a user at $page_i$ can follow from that page (that is, number of links in $page_i$) and the probability of active actions is set uniformly. Then, the probabilities are updated based on navigation of each user and learning algorithm that $L_{R\varepsilon P}$ is used in this paper. Thus, the DLA is created based on the structure of web site. The selection of action j by LA i means the activate LA j corresponds to page j. With beginning user movement from page i, LA i is activated and the corresponding action with it is given reward or penalty. Because in the model used, movements are performed only through the links on each page, the proposed algorithm does not need to check the existence of a link between the pages visited by users. Based on the web usage, data for each user -consists of pages that a user has observed since entering to the site until exit of the site- that exist in the log files, the proposed algorithm works as follows:

In the beginning of the session, for each user is identified all cycles in the current session. If there is cycle in users' navigation path, the actions in the cycles indicates the quandary of user or the dissatisfaction of user from the visited pages and must be penalized. The penalization is based on the cycle length. So, the parameter *b* which is penalization factor is calculated from Equation 7:

$$b = (\text{Length of Cycle}) * \beta \qquad (7)$$

Where β is a constant factor. Then, to each user's navigation in the current session - as regards behaviour pattern of users in the model used is the pattern of wise behaviour (that is, each user can follow the link to the page that similarity of the link with their desired information is the maximum) - based on the length of route, topic correlation of two pages (that is, Reverse of Euclidean distance between the content vector of two pages viewed (d_{ij})), and stay time of user in the page viewed, will be rewarded according to Equation 8, where l_0, d_0 and t_0 are constant factors. So, the shorter the length of the path, the more rewards given to that link and the less the Euclidean distance between two pages, namely, the more topic correlation between two pages is, consequently the more rewards are given. Also, the more stay time of user in the page viewed is, the more rewards are given.

$$a = (1/\text{length of route from } page_1 \text{ to } page_j \text{ in the current session})*l_0 + (1/d_{ij})*d_0 + (\text{stay\_time in } page_j)*t_0 \qquad (8)$$

This process is repeated for all of the sessions. After this step, the second part of the algorithm (that is, adapting DLA to user behavior) begins. In this section, using the situation of LA (active actions and their probability vector), the structure of web site will change as follows. All actions of automata, that their probability is less than a threshold (τ1), are disabled with the aim of limiting user choices in each page, to the more appropriate pages for selecting next page. In this case, if α^i_j is inactive, the connection from page i to page j is eliminated. Then, using the transitive relation if the probability of two actions from two automata connected together such as α^i_j and α^j_k, that LA^j is connected to the LA^i, both are higher than a threshold value (τ2), if α^i_k that connected to the LA^k is inactive and topic correlation between page I and page k is greater than or equal to 0.5 (to prevent excessive links on web site), is activated with the aim of reducing the number of user's navigation to achieve the desired pages. Finally, corresponds to any active action in the automata one link is inserted in the web site and corresponds to any inactive action, if exists any link, the link is eliminated. Thus, with proposed algorithm less used links and links that are not in the appropriate place are eliminated to reduce the selections and increase the accuracy of the next selections and the new routes are added for the faster guidance of users to the desired information. Pseudo code for this algorithm is shown in Figure 2.

Simulation model

Most methods reported in the area of web usage mining are used to real web usage data that has been available for researchers (such as university or college web sites). Except a few number of researches that have used data in Asuncion and Newman (2007a,

```
Procedure Web_Site_Restructuring
//N: number of pages in the web site
//DLA: a distributed learning automata contain N learning automata with changing number of actions
Begin
    For all sessions do
        Detect all cycles;
        For all traversed links in this session such as (i,j) that is in the each cycle do
            Penalize corresponding action to link j ( α^i_j ) based on eq.7;
        End
        For all traversed links in this session such as (i,j) do
            Reward corresponding action to link j ( α^i_j ) based on eq.8;
        End
    End
   //adapting DLA to user behavior
    For all actions of DLA do
        If the action probability<τ1 then
            Disable the action;
        End
    End
    For all LA of DLA do
        For all actions of LA^i do
            If the action probability of α^i_j >τ2 then
                For all actions of LA^j do
                    If the action probability of α^j_k >τ2 then
                        If α^i_k not enabled and Topic Correlation between i,k >0.5 then
                            Enable α^i_k ;
                        End
                    End
                End
            End
        End
    End
    //restructuring web site
    For all actions of DLA do
        If the action α^i_j is enabled then
            Put a link to page j in page i;
        Else
            If exist a link to page j in page i then
                Remove link to page j from page i;
            End
        End
    End
End //end of procedure
```

Figure 2. Proposed algorithm.

Table 1. Parameters in the model for experiment.

Degree of coupling	0.7
Number of agents	20000
Number of nodes	20
Number of topics	5
Tc	0.2
ΔM_t^c	-
ΔM_t^v	-
α_u	1
φ	1.2
λ	0.5
μ_m	5.97
μ_t	-
σ_m	0.25
α_p	3
σ	0.25
θ	1

b), these data are not available. In tests with such data in addition to imposition in the error of pre-processing operations, there is the possibility of influence on the algorithms used based on users' behaviour of one or more specific sites. Consequently, to evaluate the proposed algorithm used for the model introduced in Liu et al. (2004), show the structure of a hypothetical web site and how users use the web site. Validity of this model has been approved by Liu et al (2004) using the information obtained from several large web sites such as Microsoft. Accordingly, in this paper, users' interest profiles as a power law distribution, distribution content of web documents as a normal distribution and behaviour pattern of users as the pattern of wise behaviour has been considered. Other parameters used in model (Liu et al., 2004) for the simulation conducted in this paper are shown in Table 1. In this model, each web document is displayed with a content vector. Length of vector equals to the number of topics available in this web site. Each member of this vector shows the degree of relationship between corresponding document to that vector and one of these topics.

$$C_n=\{CW^1_n, CW^2_n, \ldots, CW^m_n\} \qquad (9)$$

Model introduced in Liu et al. (2004) is only able to make sessions of users and time stamp of each page in the user navigation path and is not clear. Therefore, in this study, the model is changed so that stay time in each page of the user's navigation path also can be produced.

Evaluation metric

To evaluate the proposed algorithm with other methods, correlation measure is used. This metric can be determined by linear correlation between the data set. Correlation for two data sets P and P ' is calculated according to Equation 10, where N is number of data sets, P is structure generated by the simulation model in the ideal state and the P' is structure created by the proposed algorithm.

$$Corr(P,P') = \frac{\sum PP' - (\sum P \sum P')/N}{\sqrt{\sum P^2 - (\sum P)^2/N}\sqrt{\sum P'^2 - (\sum P')^2/N}} \qquad (10)$$

$$p=\{p_{ij}|i,j=1,2,3,\ldots,n, i\neq j\} \qquad (11)$$

$$p_{ij}=\frac{d_{ij}^{-1}}{\sum_{k=1}^{m} d_{ik}^{-1}} \qquad (12)$$

$$d_{ij}=\sqrt{\sum_{k=1}^{m}\left(cw_i^k - cw_j^k\right)} \qquad (13)$$

$$p'=\{p'_{ij}|i,j=1,2,3,\ldots,n, i\neq j\} \qquad (14)$$

dij is Euclidean distance between two documents that is calculated according to Equation 13.

SIMULATION RESULTS

In this simulation, parameters are considered according to Table 1. Also, all the tests on the proposed method and the method presented in Baradaran and Meybodi (2007b) (Hashemi algorithm) are done based on the same series of sessions (web usage data).

Test 1

To demonstrate the efficiency of the proposed algorithm, the degree of coupling defined in the model (r = 0.7) is considered less (r=0.5). Consequently, number of internal links of web sites produced will be less than desirable level. With running of the proposed algorithm on usage data from this web site, the proposed method can detect the removed links (than the value of r = 0.7). As shown in Figure 3, with the proposed algorithm for different values of τ_1={0.01, 0.02, 0.03, 0.04, 0.05}, the number of eliminated links rather than ideal state, in comparison with existing method in Baradaran and Meybodi (2007b) (Hashemi algorithm) and primary web site is less.

Test 2

A comparison of the efficiency of the proposed algorithm with the only reported method based on DLA is used for the correlation metric. Figure 4 show that the proposed algorithm has higher correlation. It means that more convergence of the structure is obtained by using the proposed method to the optimal structure (r=0.7), than the method in Baradaran and Meybodi (2007b) (Hashemi algorithm).

Test 3

In this experiment, we used the structure obtained by the proposed method in the model and checked the behaviour of the users in the new structure. According to Figure 5, the number of users' requests has been reduced in comparison with method in Baradaran and

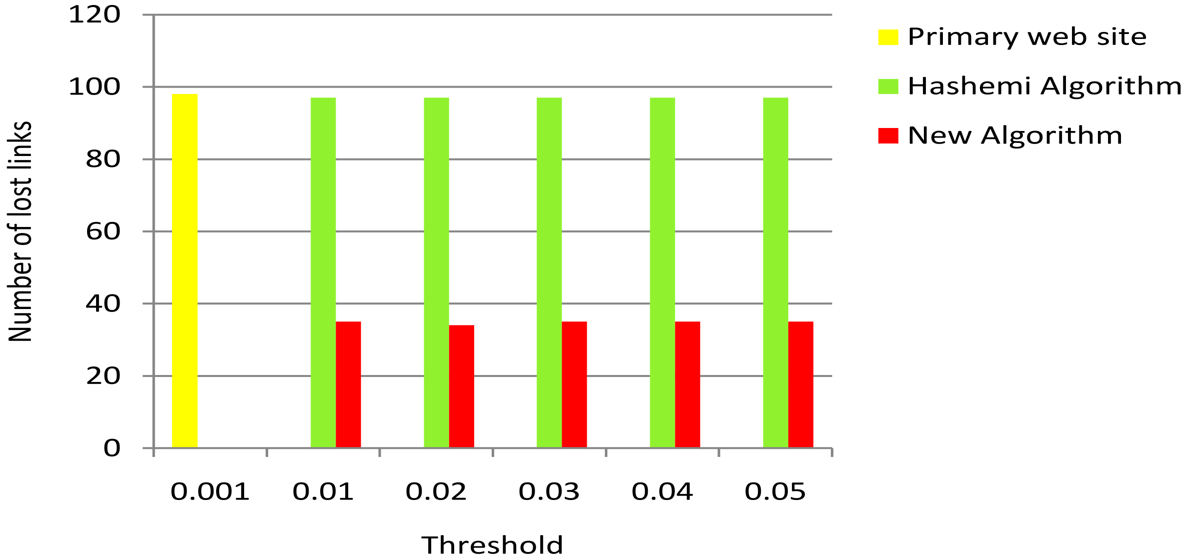

Figure 3. The number of eliminated links rather than ideal state, in comparison with Hashemi algorithm and primary web site with different values τ_1 و $\tau_2 = 0.5$

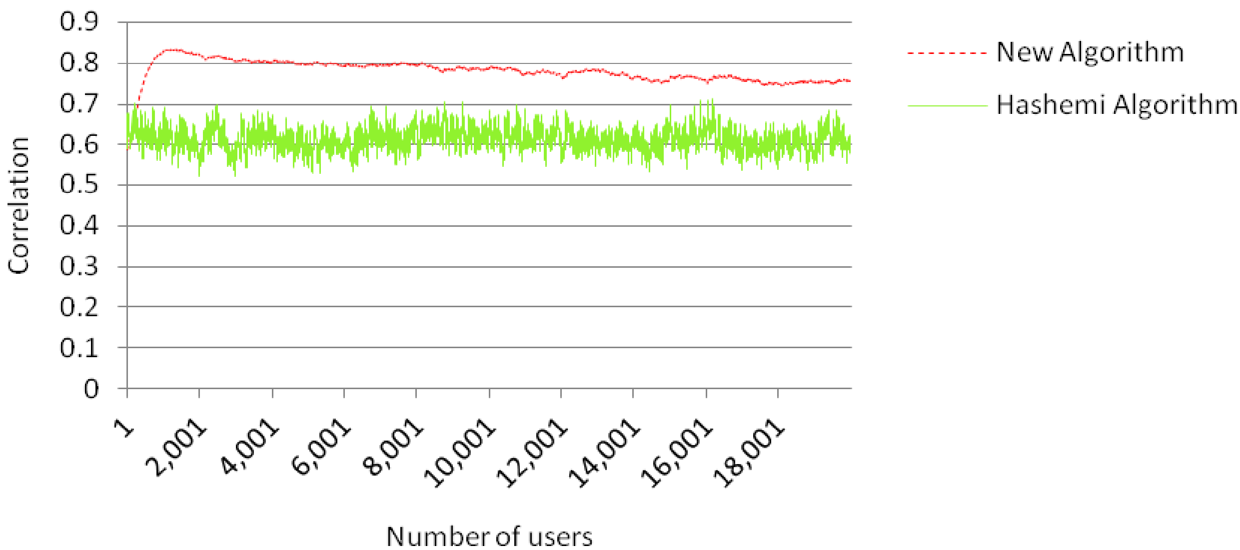

Figure 4. Comparison between correlation of New algorithm and Hashemi algorithm.

Meybodi (2007b) (Hashemi algorithm) and primary web site. This reduction represents faster access of users to the information, increase their satisfaction, increase efficiency of the web site and reduce the use of the web server resources.

CONCLUSIONS AND FUTURE WORKS

Adaptation is an approach that focuses on the bases of a web site design such as the ease of use and navigation in the site. As users' view often differs from what web site designers have in mind about site structure, web mining techniques can play an important role in helping the designers to design web site with optimal structure so that users can get their desired information quickly. Most existing methods can only offer a few new links based on the first page and the last page in the users' navigation path without considering intermediate pages. Method presented in this paper using DLA and interactions of

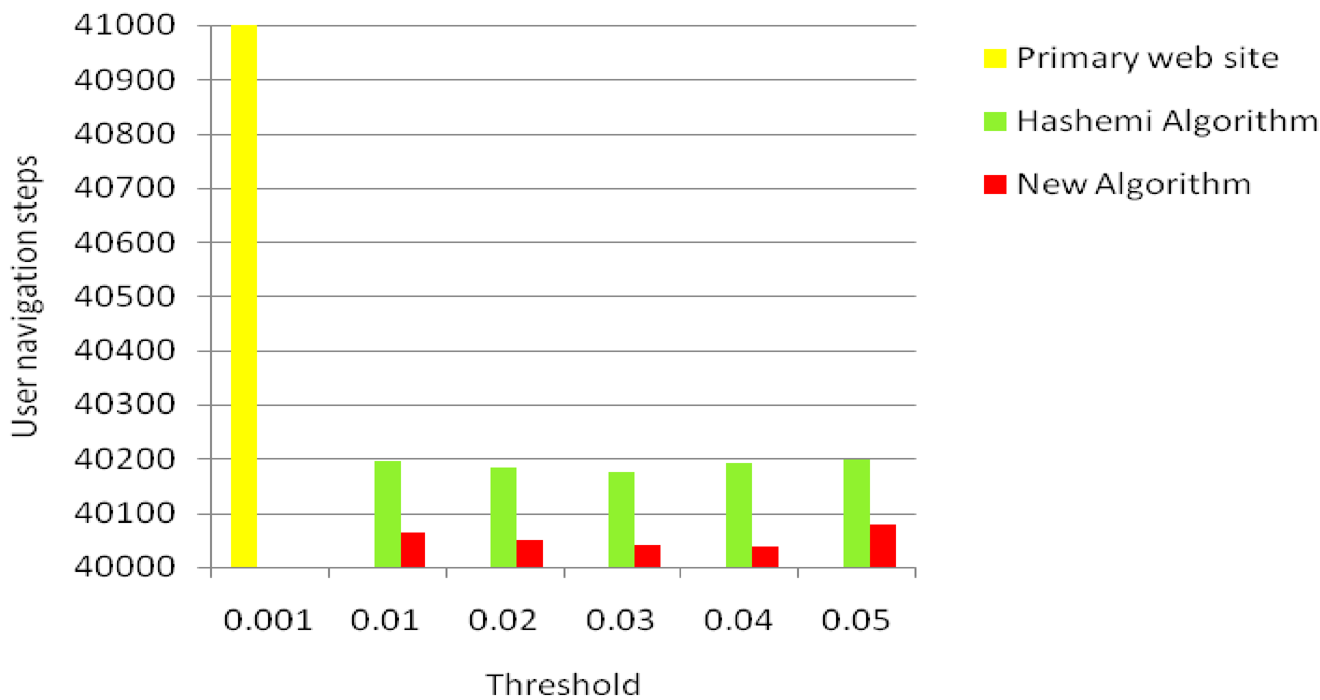

Figure 5. The number of users' requests in web site, compared with Hashemi algorithm and primary web site with different values τ_1 و $\tau_2 = 0.5$.

users in the past, checked all the page viewed in the users' navigation path and provides the new structure for web site based on the length of route, topic correlation of two pages and stay time of user in the page viewed and also considered a restriction to prevent excessive links on website based on the topic correlation of the pages. so that in this new structure, users can achieve their desired information with less number of steps and consequently, increases users' satisfaction and efficiency of web sites. Simulation results showed that the proposed algorithm creates a new structure for the site studied that is closer to the optimal structure. As a result, the efficiency of the proposed algorithm in comparison with only reported method based on DLA is higher and due to considering the variable learning parameter, does not need to readjust the learning parameters if there is change in the number of web pages. In future works, our aim will be the use of the combination of LA and Bayesian networks in order to use the inference capabilities of Bayesian networks and also the use of Semantic web concepts to improve the web site.

REFERENCES

Abedin B, Sohrabi B (2009). Graph theory application and web page ranking for web site link structure improvement. Behav. Inform. Technol., 28(1): 63-72.

Anari B, Meybodi MR (2007). A Method based on distributed learning automata for determining web documents structure. in 12th International CSI Computer Conference (CSICC'07). Tehran. Iran. pp. 2276-2282.

Asuncion A, Newman DJ (2007a). Microsoft anonymous dataset. in UCI Machine Learning Repository: University of California. Irvine. School Inform. Comput. Sci., http://kdd.ics.uci.edu/databases/msweb/msweb.html.

Asuncion A, Newman DJ (2007b). MSNBC Dataset. In UCI Machine Learning Repository: University of California. Irvine. School Inf. Comput. Sci., http://kdd.ics.uci.edu/databases/msnbc/.

Baradaran HA, Meybodi MR (2007a). Web Usage Mining using Distributed Learning Automata. In 12th International CSI Computer Conference (*CSICC'07*), Tehran, Iran. pp. 553-560.

Baradaran Hashemi A, Meybodi MR (2007b). Adaptive Web Sites Using Learning Automata. In 1th International Data Mining Conference (*IDMC*), Tehran, Iran, pp. 1-9.

Beigy H, Meybodi MR (2002). A New Distributed Learning Automata Based Algorithm For Solving Stochastic Shortest Path Problem. Proceedings of the Sixth International Joint Conference on Information Science. Durham, USA, pp. 339-343.

Beigy H, Meybodi MR (2006). Utilizing Distributed Learning Automata to Solve Stochastic Shortest Path Problem. Int. J. Uncertainty, Fuzziness Knowledge-based Syst., 14(5): 591-617.

Brickell J, Dhillon IS, Modha DS (2007). Adaptive website design using caching algorithm. Adv. Web Min. Web Usage Anal., Springer, 4811: 1-20.

Chitraa V, Davamani AS (2010). A survey on preprocessing methods for web usage data. IJCSIS, 7(3): 78-83.

Forsati R, Meybodi MR, Mahdavi M (2007). Web Page Personalization based on Distributed Learning Automata. Proceedings of the Third Information and Knowledge Technology. Ferdowsi University of Mashad, Mashad, Iran, pp. 27-29. Nov.

Kosala R, Blockeel H (2000). Web mining research: A Survey. ACM SIGKDD, pp. 1-15.

Lee JH, Shiu WK (2004). An adaptive website system to improve efficiency with web mining techniques. Adv. Eng. Inform., Elsevier, 18: 129-142.

Lin CC (2006). Optimal Web site reorganization considering information overload and search depth. Eur. J. Oper. Res., 173(3): 839-848.

Liu J, Zhang S, Yang J (2004). Characterizing Web Usage Regularities with Information Foraging Agents. IEEE Trans. Knowl. Data Eng., 16(5): 566-584.

Meybodi MR, Beigy H (2001). Solving Stochastic Path Problem Using Distributed Learning Automata. Proceedings of The Sixth Annual International CSI Computer Conference, CSICC2001 Isfahan, Iran, pp. 70-86.

Mobasher B, Cooley R, Srivastava J (1999). Creating Adaptive Web Sites through usage based clustering of URLs. IEEE Knowl. Data Eng. Workshop (KDEX'99), pp. 1-7.

Narendra K, Thathachar MAL (1989). Learning Automata: An Introduction. Prentice Hall, Englewood Cliffs, New Jersey, p. 476.

Perkowitz M, Etzioni O (1997a). Adaptive sites: Automatically learning from user access patterns. Technical report, Department of Computer Science and Engineering, University of Washington, pp. 1-11.

Perkowitz M, Etzioni O (1997b). Adaptive Web Sites: an AI Challenge. In Fifteenth Int. Joint Conf. Artif. Intell., (IJCAI 97), pp. 16-23.

Perkowitz M, Etzioni O (1999). Adaptive Web Sites: Conceptual Cluster Mining. In 17th International Joint Conference on Artificial Intelligence (IJCAI '99), Stockholm, Sweden, pp. 264-269.

Perkowitz M, Etzioni O (2000a). Adaptive Web Sites. Comm. ACM., 43(8): 152-158.

Perkowitz M, Etzioni O (2000b). Towards adaptive Web sites: conceptual framework and case study. Artif. Intell., Elsevier, 118(1-2): 245-275.

Saati S, Meybodi MR (2006). Document Ranking Using Distributed Learning Automata. Proceedings of 11th Annual CSI Computer Conference of Iran. Fundamental Science Research Center (IPM). Computer Science Research Laboratory, Tehran, Iran, pp. 467-473. Jan. 24-26.

Shahabi C, Kashani FB (2003). Efficient and Anonymous Web-Usage Mining for Web Personalization. INFORMS J. Comput., 15(2): 123-147.

Shokry RA, Saad AA, El-Makkey NM, Ismail MA (2006). Using New Soft Clustering Technique in Adaptive Web Site. In IEEE/WIC/ACM Int.Conf. Web Intell. Intell. Agent Technol., pp. 281-286.

Suneetha KR, Krishnamoorthi R (2009). Identifying User Behavior by Analyzing Web Server Access Log File. Int. J. Comp. Sci. Netw. Secur. (IJCSNS), 9(4): 327-332. April.

Tao YH, Hong TP (2008). Web usage mining with intentional browsing data. Sci. Direct, 34: 1893-1904.

Thakare SB, Gawali SZ (2010). A effective and complete preprocessing for web usage mining. Int. J. Comp. Sci. Eng., 2(3): 848-851.

Thathachar MAL, Bhaskar RH (1987). Learning automata with changing number of actions. IEEE Trans. Syst. Man Cybernet., 17(6): 1095-1100.

Tiwari RG, Husain M, Gupta S, Srivastava AP (2010). Recuperating web sites link structure using fuzzy relations between the content and web pages. Int. J. Comput. Appl., 1(11): 34-39.

Tyagi NK, Solanki AK, Tyagi S (2010). An algorithmic approach to data preprocessing in web usage mining. Int. J. Inf. Technol. Knowl. Manage., 2(2): 279-283. July-Dec.

Weigang L, Dib MVP, Teles WM, Andrade VMD, Melo ACMAd, Cariolano JT (2002). Using ant's behavior-based simulation model AntWeb to improve website organization. In SPIE's Aerospace/Defense Sensing and Controls Symposium: Data Mining, Orlando, USA, pp. 229-240.

Zhu J, Hong J, Hughes JG (2002). Using Markov Chains for Link Prediction in Adaptive Web Sites. Springer, pp. 60-73.