A novel web page recommender using data automatic clustering and Markov process

Fereshteh Darbandi Monfared¹

Received: 2 July 2019 / Accepted: 18 November 2019 / Published online: 28 November 2019
© Springer Nature Switzerland AG 2019

Abstract

As the web expands and data grow in the networks, finding useful knowledge from the World Wide Web becomes a major challenge. As intelligent systems, the recommender systems help users to find their favorite resource among a large body of information. This research aimed at providing a new page recommender system, which enhances the accuracy of the suggested pages based on the user interest, Automatic Clustering-based Genetic Algorithm (ACGA), and all modified 3th-order Markov Model. Therefore, the frequency and page visiting time by each user have been extracted from the log file of the web server, while the user interest in the page during a session has been computed by a linear combination. ACGA involved 2 phases of the cluster based on (1) similarity measure and (2) Genetic Algorithm. The novel cluster partitioned the vectorization sessions in separate clusters through a new fitness function. Moreover, all modified 3th-order Markov Model used all orders of Markov Model simultaneously, and predicted the next page in the page recommender system. The research did these experiments on a real CTI dataset. Experimental results and their comparison with other approaches showed superiority in the accuracy of the proposed system.

Keywords Recommender system (RS) · User interest · Automatic clustering · Genetic algorithm · Markov process

1 Introduction

With regard to the huge volume of information resources, the World Wide Web is considered as a great environment to distribute information. According to the studies conducted so far, information overflow takes place when a search is done by a browser. In addition, information management is provided by the recommender systems (RSs) for user's access. Results showed that RS is a tool that offers useful products among many possible options to the users [25]. One of the most fundamental goals of the recommender systems is to collect various information about the user's interests and items in the system. It has been found that there are different resources and methods for gathering such information, including explicit and implicit methods [19]. According to the results, the user explicitly

expresses their interests in the explicit method of information collection. Put differently, an implicit method is slightly difficult; that is, the system should find the users' interests by controlling and following their behaviors and activities [5]. Some researchers divided RSs into 5 main categories of content-based techniques, collaborative filtering techniques, demographic filtering techniques, knowledge-based techniques, and hybrid filtering techniques [2]. As reported by the researchers, these systems are used by most large companies such as Facebook and Google [28].

It is widely accepted that clustering is an algorithm with the highest utilization in the RS [10]. Therefore, clustering technique is applied to cluster data session into a number of similar clusters, which influences the RSs performance. In a web browser, the users' activities are expressed as a

✉ Fereshteh Darbandi Monfared, f.darbandi.monfared@gmail.com | ¹Department of Computer, Bushehr Branch, Islamic Azad University, Bushehr, Iran.

session in order to navigate the desired pages. Hence, a user session is comprised of a number of pages that show the user's sequential pattern. Modeling the users' session is performed by clustering. A lot of RSs have been provided, which use k-means and modified K-means technique for clustering sessions [1, 8, 20, 23, 30–32]. According to the researchers, these algorithms require the number of clusters K as an input. Then, they randomly choose K number of initial seeds (cluster centers) from a dataset. However, it can be difficult to guess the number of cluster K in a big dataset with no prior knowledge. Moreover, it has high sensitivity to the quality of initial seeds. Therefore, this research proposed a novel clustering method in order to find the number of clusters automatically, and used high quality initial seeds. A main contribution of the novel RS includes vectorization of web user sessions using a linear combination. Web server log files have important information about user's surfing on the web, and the research used frequency and duration of viewing pages by the user for calculating the users' interest. Another major contribution of the RS is the clustering session based on GA called ACGA, which is capable of automatically finding clusters. As an optimization problem, ACGA intends to find the best value for the validation index and yields the most acceptable result. With the help of a novel fitness function, this research intended to find clusters with high similarities. This method could be applied as a fitness function and cluster evaluation technique. As it is known, the fitness function regards a clustering solution that has compact clusters and big separations or gaps among the clusters. However, since GA spends many times for running big data, this research proposed a cluster-based similarity for finding initial cluster to resolve the problem. The GA uses the average data of each cluster as the input. However, the output of the ACGA consists of a number of clusters that do not provide information about the probability of viewing pages in the future. Thus, it is necessary to use discovery models of sequential patterns, including Markov Model (MM) to predict the web pages. Markov models represent a method for discovering sequential patterns. Researchers have used various orders of MM for predicting the next pages in RSs [21, 30]. They found that MMs (1st- or 2nd-order) have low coverage and accuracy in prediction. Moreover, high orders require a large memory and have numerous states. All k -th order MM overcomes these problems. Authors [7, 16, 22] used this model in the prediction phase but they did not use different orders of MM simultaneously. Hence, the present research used a modified all k -th order MM for increasing the prediction accuracy by taking advantage of all orders of MM concurrently. This method highlighted the probability of the same pages in all order of MM.

This research compared its novel RS with Harmony Session Clustering Recommender (HSCR) [8], Harmony K-means Session Clustering Recommender (HKSCR) [8], Interleaved Harmony K-means Session Clustering Recommender (IHKSCR) [8], K-means, all k -th order MM (KMM), K-means, MM, and Popularity and Similarity-based Page Rank (KMMPSPR) [30] on CTI dataset for 3 evaluation criteria called accuracy, coverage, and F-measure. The proposed RS has better performance than other techniques in CTI dataset for evaluating the criteria. Moreover, the results of the ACGA have been compared with the Automatic Clustering-based on Differential Evolution (ACDE) [29] and K-means [14] algorithms through the Chou-Su (CS) [4] and novel cluster evaluation criteria of Compactness and Separation Measure of Clustering (CSMC), which indicated that it has a high performance.

Section 2 reviews and analyzes the previous studies in the field. Section 3 deals with the proposed algorithm and its steps. Section 4 discusses time complexity of the proposed algorithms. Section 5 evaluates and analyzes the output of the proposed recommender system. Section 6 concludes the research and provides a number of suggestions for further research.

2 Literature review

Clustering the web sessions is one of the important parts in the web usage mining, which is a technique to discover clusters from the web data. Sessions in web are represented in two vector and non-vector models. Many authors used vector model by binary representation for web sessions, while several web session clustering used non-vector model for clustering sessions. Discovering sequential patterns enclosed in the sessions has been used to predict the probability of the page viewing by various Markov models. The main goal of this research has been to propose a non-vector model for sessions. Since the frequency and page visiting time are connotations of the users' interest, their combination could affect the recommendation accuracy. Another main objective of the research has been to design an automatic clustering session and use a modified all k -th order MM for predicting pages. To cluster sessions, ACGA method has been used for clustering through ACGA as an optimization problem that aimed to find an optimal part of the solution space, which optimized function and cluster sessions. Each solution in the population had a fitness value that relied on the function to be optimized. This research took the cluster evaluation criterion as fitness function. The CSMC method has been defined as a new validation index, and covered both compactness and separation factors in the clusters. As a sequential decision process, the recommendation process

should be implemented by modifying all k -th order MM. This MM combined probability pages in different orders of MM and presented a recommendation with high probability. Here, the research provides a literature review of the web page recommender systems, clustering techniques based on the metaheuristic algorithm, and Markov model.

2.1 Web page recommender systems

The RS is a kind of decision-support and intelligent system. It could be used for suggesting the web pages to be visited next in a big website in order to guide the web users to find the relevant information they needed. This type of RS is called web page recommender system. These systems used various techniques for recommendation of pages for a user.

There are numerous studies related to the recommender systems that employed K-means algorithms for clustering sessions [1, 24, 31, 32]. Researches used them because of their simplicity; however, they had generally high sensitivity to the number of clusters and initial seeds in a huge dataset.

Selvi et al. [27] proposed a new RS that originates the Collaborative filtering approach in "A novel optimization algorithm for recommender system using modified fuzzy c -means clustering approach". The rating users have been clustered with minimal error rate using a proposed modified fuzzy c -means (MFCM) clustering approach. Users in the cluster have been further optimized through the proposed modified cuckoo search (MCS) algorithm with less number of iterations. Using the combining MCS with MFCM, the presented RS reduced the recommendation error rate and provided a list of recommendations with high accuracy.

In their research under "Web user session clustering using modified K-means algorithm", Poornalatha et al. [23] presented a new clustering method aiming at the clustering web users' sessions. Since the users' sessions had variable lengths, a new distance criterion defined as the Variable Length vector Distance (VLVD) has been defined. The proposed method modified the K-means method for clustering web sessions. This approach showed that the number of the iterations of this algorithm is less than that of the regular K-means, and the sessions are placed in appropriate clusters according to the similarity between them. However, two problems of the K-means technique have been not considered in this research.

Mishra et al. [17] conducted a study titled "A web recommendation system considering sequential information" and presented a recommender system using the similarity functions of Sequence and Set Similarity Measure (S^3M) and Singular Value Decomposition (SVD). After identifying different users, user clicks have been created by separate

sequences, which have been then attained through the S^3M similarity function. Upon the entrance of a new user, the first M cluster with the most similarity has been found, and then the response matrix has been formed according to the clusters. The weight vector for the current user has been created according to the location of its pages. Before the page suggestions, the SVD algorithm first reduced the size, after which the pages with the highest weight have been displayed by the output of this step. Moreover, Mishra et al. considered sequential information in web navigation along with content information.

They published the "Web page access prediction based on an integrated approach" and provided a recommender system based on KMMPSPR [30]. The main idea in this research has been to solve the problem of the Markov model in providing suggestions when pages had the same probability. After clearing data and identifying users and sessions, this algorithm used the k -means algorithm for clustering sessions. Then, 1st- and 2nd-order Markov model has been applied on each cluster. Here, when the pages identified by the Markov model were likely to be identical PSPR applied to the proposed one, the algorithm decided which one to propose based on their rank. The limitation of this research is that, it considers only low orders of Markov model for predicting pages.

KMM is a method, which uses K-means and all-3th order MM for recommendation. In this algorithm, the user session is initially clustered by the K-means algorithm. Then, MMs 1 to 3 are applied to each of the clusters. Afterwards, in the prediction phase, the first cluster is determined upon the entrance of a new user, and the pages are suggested using the Markov model with all 3th order. This model for vectoring sessions uses only frequency. Moreover, the number of the clusters is defined by users as an input in K-means, but novel RS uses information in log file and does not need defining the number of clusters.

In their paper titled "An effective web page recommender using binary data clustering", Forsati et al. [8] proposed a new recommender system called harmonic session clustering. The main contribution of this research is clustering user sessions as zero and one vectors using harmony search optimization algorithm. Algorithms such as HSC, HKSC, and IHKSC have been presented in this research. In the HSC, users' sessions have been clustered by harmony search algorithm. The objective function has been the error least-squares in the HSC. In HKSC, the centers have been introduced as problem solutions, and the K-means algorithm has been implemented on each of them. The algorithm ultimately represented the best response as the top solution. The IHKSC consisted of two stages. Firstly, clustering has been performed by the HSC algorithm after N iterations. Then, the best generation, which is superior over others, as cluster centers, has been taken

from a harmonic algorithm and introduced as an input to the K-means algorithm, which performed the clustering. The criterion for assessing the clustering system has been the Average Distance of Sessions to the Cluster (ADSC) and Visit-Coherence (VC). The criteria for evaluating the recommender system included accuracy, coverage, and F-measure. In each algorithm, the number of the clusters has been considered as input and they did not use information in the log file and instead only considered whether a page has been seen by the user or not.

Selvi et al. provided "*Personal recommender system based on user interest community in social network model*" as a novel time weighted score matrix, in which the users and items with higher correlation are clustered into the same community by using differential equations [3]. Firstly, users' interest is calculated based on the rounding-Forgetting. Time and score matrix are used as inputs in this function. Secondly, difference equation is performed to follow the clustering evolutionary process and cluster the users and items into several communities. Finally, a recommendation list is proposed according to the predicted scores. Results indicated that the system could improve the recommender system based Collaborative filtering.

2.2 Clustering technique based on Metaheuristic algorithm

Maulik et al. [15] published "*Genetic algorithms-based clustering technique*" and designed a genetic algorithm for clustering data. They found that the number of the clusters as input is received from the user. Moreover, the K cluster centers have been randomly chosen from dataset. Each chromosome in the population represented the center of clusters. The researchers used inverses of Sum Squared Error (SSE) for Fitness function. This method could be run in artificial and real datasets, and has been compared by K-means algorithm, which showed more acceptable results.

Lin et al. [11] reported "*An efficient GA-based clustering technique*", and proposed a GA-based clustering technique, which chooses cluster centers directly from the dataset so that the number of clusters is defined by the user. According to these researchers, the length of the chromosome equals to the size of the dataset, in which the i th gene of the chromosome represents the i th data point in the dataset. Therefore, for a data point of index i to be the candidate for the center of a cluster, the value of the corresponding i th gene in the chromosome is set to "1"; otherwise, it is set to "0". Thus, each chromosome in the population is evaluated by inverses of Davies–Bouldin (DB) measure as the fitness function. This algorithm was tested on an artificial dataset.

Nonetheless, limitations in the "*Genetic algorithms-based clustering technique*" by Maulik et al. [15] and "*An efficient GA-based clustering technique*" by Lin et al. [11] include their requirement for a user input on the number of clusters and random selection of the initial seeds of dataset.

Rahman et al. [26] presented a new clustering technique in "*A hybrid clustering technique combining a novel genetic with K-means*". This technique is a combination of k-means and GA together called GenClust technique. It aimed at reaching more acceptable quality clusters without any necessity for user inputs for the number of clusters. The genes are found by genetic algorithm both randomly and deterministically in the initial population. GenClust has been successful to automatically find the right number of clusters and identify the right genes via a novel initial population determination approach. To achieve even a higher quality clustering resolution, centers have been given K-means as starter seeds permitting the adjustment of initial seeds according to the requirements.

GenClust suffers from the following limitations. Firstly, its initial population selection procedure has a time complexity of $O(n^2)$, which may be problematic for large datasets. Secondly, it uses a set of user defined radii of clusters to obtain the initial population. It is widely accepted that actual clusters possibly have radii values that vary from one dataset to another that is likely dependent on several factors, including the dataset dimension.

Swagatam et al. [29] developed a novel automatic clustering called ACDE in their study titled "*Automatic clustering using an improved differential evolution algorithm*". This method employs an activation threshold (control genes) in the range between 0 and 1 for active or nonnative centers. Such an idea finds the number of the clusters automatically. Moreover, the cluster centroids are randomly fixed between $X_{\max}$ and $X_{\min}$, which implies the highest and lowest numerical values of any property of the dataset under test. Furthermore, the response applied DB and CS measures for evaluation. According to the results, this method is capable of finding the number of clusters in various runs; however, accuracy in the clusters would not be perfect.

Thus, for evaluating the clustering strategies, the analysis would be done for standard dataset, namely Wifi-localization.¹

Figure 1 depicts a visual view of the performance of 4 clustering techniques for the Wifi-localization dataset.

¹ Standard dataset. <http://storm.cis.fordham.edu/~gweiss/data-mining/datasets.html>.

Fig. 1 **a** Three dimensional plot of the unlabeled Wifi-localization dataset using the first three features. **b** The labeled Wifi-localization dataset. Clustering of Wifi-localization dataset by **c** GA based SSE, **d** GA based DB, **e** GenClust, and **f** ACDE

Table 1 Comparison of clustering methods for Wifi-localization dataset

Clustering technique	Number of cluster	Initial center of cluster	Execution time (s)	Accuracy (%)
GA based SSE [15]	Defined by user	Randomly	18	66.15
GA based DB [11]	Defined by user	Randomly	31	57.45
GenClust [26]	Automatic	Randomly and deterministic	9054	72.15
ACDE [29]	Automatic	Randomly	154	56.60

Table 1 compares the outputs of various clustering techniques on the basis of experimental tests.

2.3 Markov model

The Markov models are robust mathematical ones that have been suggested to discover sequential patterns, study, and understand random procedures that have also been useful in the prediction of the web pages. Nonetheless, low-order (1st or 2nd order) Markov models cannot precisely predict the next page, which will be visited by the user, because they do not deal with the user's history deeply enough. Put differently, disadvantages of higher

order Markov models are a large number of states and low coverage.

"Mining longest repeating subsequence to predict World Wide Web surfing" has been reported by Pitkow et al. They employed all k-th order MM to resolve the problem of low-order MM [22]. This technique applies higher order model for prediction. If this order is not able to cover this process, it will be repeated for lower order. The model does not utilize all orders of information simultaneously.

Moreover, Mamoun et al. [16] designed a new modified MM in their study titled "Prediction of user's web-browsing behavior: application of markov model" where all sessions consist of the same pages, but they are considered the same on different order. The researchers argued that an

Fig. 2 Flowchart of the proposed algorithm in the offline phase **Fig. 3** Flowchart of the proposed algorithm in the online phase

action on the web may be performed by various paths irrespective of the order that users select. In addition, they decreased the prediction model dimension via elimination of the sessions, in which the pages would be iterated. This research declined the size of the model with no direct effects on the model accuracy.

In addition, Dhyani et al. published an article titled “Modelling and predicting web page accesses using markov processes”. These researchers assumption has been that the probability of viewing pages would not modify with time in MM [7]. Therefore, the matrices by 2nd until k-order would be achieved by exponent a first-order matrix, after which they employ the all kth-order MM for suggesting pages for the user. Dhyani et al.’s study suffered from a major problem, which is related to the calculation method of probability. Moreover, they did not consider the viewing pages sequence in their research.

3 The proposed algorithm

The suggested technique contains 2 offline and online phases. During the offline phase, the pre-processing operations will be initially accomplished on the web server’s registry files for identifying the users and sessions. Afterwards, vectorization of the web users’ sessions would be performed to realize the user’s interest in web pages. Then, the vectored sessions would be clustered via automatic clustering on the basis of the clustered similarity and genetic algorithm. Moreover, Markov model would be used to each of the clusters. During the online stage, the pages would be provided with the arrival of a new user session, in which the prediction action would be conducted. Figures 2 and 3 portray the structure of the suggested algorithm in 2 offline and online phases.

3.1 Data pre-processing

According to the algorithm, web server logs have been regarded as one of the main sources of information to extract information in the field of web mining. Extensive studies have been considered preparing data for initial processing, collecting, and integrating these data sources for different analyses. However, preparing the applied data would have certain challenges resulting in algorithms and rational techniques for initial processing, including data combination and clearance, user and sessions identification, as well as the viewed page identification [6]. After clearing the data, identification of the users would be the most significant step in the data processing. When the users have been identified, it is necessary to identify the user sessions. According to some researchers, a user session is a set of pages visited by users during a certain visit to the website [9].

3.2 Making session vectors

If $P=\{p_1, p_2, \dots, p_n\}$ is a set of the pages available by the users of a website, and each page is associated with a unique URL, then $S=\{s_1, s_2, \dots, s_n\}$ is a set of web users’ sessions and every $s_i \in S$ is a subset of P . Each session (s_i) is represented by a n-dimensional vector as $s_i=\{w(p_1, s_i), w(p_2, s_i), \dots, w(p_n, s_i)\}$, where $w(p_j, s_i)$ represents the weight defined for the j -th web page visited in the session s_i .

For weighing the pages, the users’ interests should be determined. Therefore, researchers utilized the ‘page frequency’ and ‘visit duration’. The *page frequency* refers to the number of visits to a particular web page (*page*), whereas the *visit duration* shows the time spent on a given page by a user (*user*) [12]. The parameter *Duration* represents the time spent on the page. Also, *size* implies the size of the

page in bytes, and *number of Visits* refers to the repetition of the page in one session [18, 34]:

$$\text{Frequency}(\text{user}, \text{page}) = \frac{\text{number of visits}(\text{page})}{\sum_{\text{page} \in \text{visited pages}} (\text{number of visits}(\text{page}))} \quad (1)$$

$$\text{Duration}(\text{user}, \text{page}) = \frac{\frac{\text{duration}(\text{page})}{\text{size}(\text{page})}}{\text{Max}_{\text{page} \in \text{visited pages}} \left(\frac{\text{duration}(\text{page})}{\text{size}(\text{page})} \right)} \quad (2)$$

Equation (3) has been employed for merging the *Frequency* and *Duration* parameters into a single weight. λ can be adjusted and is attainable by experimentation:

$$\text{Interest}(\text{user}, \text{page}) = \lambda * \text{Frequency}(\text{user}, \text{page}) + (1 - \lambda) * \text{Duration}(\text{user}, \text{page}) \quad (3)$$

Ultimately, the identified sessions would be divided into 2 sets of ‘training sessions’ and ‘test sessions’. The training sessions set is employed to implement the suggested method, whereas the test sessions set is applied to evaluate it.

3.3 Automatic web session clustering

The ACGA is a GA for clustering sessions on the web. In general, this algorithm contains 2 phases so that clustering-based similarity is implemented in the first phase.

During the first phase, data with the highest similarities establish one cluster. The data in each cluster connects with each other directly or indirectly. This technique is on the basis of the link structure for clustering data. After discovering the first cluster in the first step, this research employs these clusters as input data for GA in the second phase. Thus, the clusters are combined in the second step. It should be noted that the chromosomes structure would be on the basis of a matrix. However, an objective function is used to assess the chromosomes quality. This condition returns the fitness (or cost) values of the chromosomes, followed by the selection of the percentage of the chromosomes with the maximum fitness or minimum cost function values for the next iteration. In general, the genetic operators of crossover and mutation are used to the chromosomes for achieving a more acceptable population. Therefore, the crossover and mutation operations would be iterated till the repetition finals are satisfied. Thus, at the resulting iteration of the algorithm, the fittest chromosome would be returned as the best found solution. Figure 4 indicates the structure of the recommended clustering algorithm.

3.3.1 Clustering-based similarity

The present research presumes that a dataset D possesses a set of records $R = \{R_1, R_2, \dots, R_n\}$, and each record has a set of attributes $A = \{A_1, A_2, \dots, A_m\}$. Equations (4) and (5) are used to achieve the distance and similarity between

the data:

$$d(R_i, R_j) = \sqrt{\sum_{k=1}^m (R_{ik} - R_{jk})^2} \quad (4)$$

$$\text{sim}(R_i, R_j) = \frac{1}{1 + d(R_i, R_j)} \quad (5)$$

Euclidean distance in Eq. (4) has been applied for distance, where m stands for the number of dimensions or the number of features. Each data selected a set with the maximum similarities. However, data with direct or indirect inter-connections would be merged. Correspondingly, the clustering-based similarity would be initially identified. For explaining the above approach, assume a dataset with 20 records and 4 attributes. Table 2 reports this dataset. When the distance and similarities between every piece of data have been computed, the nearest data would be discovered. Table 3 presents this step.

As seen in the Table 3, each data is clustered with another data with the greatest similarity. In addition, the clusters would be checked if the same members exist between them; and thus they are combined. This procedure would be continued till the clusters do not have the same members. The procedure acts on the basis of intersection and union. For instance, R_4 has high similarity to R_3 and vice versa. In the first step of clustering, there are just two similar members, but members similar with this cluster would be obtained while searching in other clusters, $\{R_7, R_3\}$ and $\{R_9, R_4\}$ that have the same members with $\{R_3, R_4\}$. They are combined with each other and put in one cluster $\{R_3, R_4, R_7, R_9\}$. Finally, this cluster is compared with other clusters and similar data would be found. Cluster $\{R_{14}, R_9\}$ have intersection with this cluster, so final cluster would be $\{R_3, R_4, R_7, R_9, R_{14}\}$. This process would be implemented for each cluster in order to find the final clusters. Figure 5 indicates final cluster that finds based similarity technique.

Fig. 4 Flowchart of the proposed clustering algorithm **Table 2** A sample dataset

	A1	A2	A3	A4		A1	A2	A3	A4
R1	5.1	3.5	1.4	0.2	R11	5.4	3.7	1.5	0.2
R2	4.9	3	1.4	0.2	R12	4.8	3.4	1.6	0.2
R3	4.7	3.2	1.3	0.2	R13	4.8	3	1.4	0.1
R4	4.6	3.1	1.5	0.2	R14	4.3	3	1.1	0.1
R5	5	3.6	1.4	0.2	R15	5.8	4	1.2	0.2
R6	5.4	3.9	1.7	0.4	R16	5.7	4.4	1.5	0.4
R7	4.6	3.4	1.4	0.3	R17	5.4	3.9	1.3	0.4
R8	5	3.4	1.5	0.2	R18	5.1	3.5	1.4	0.3
R9	4.4	2.9	1.4	0.2	R19	5.7	3.8	1.7	0.3
R10	4.9	3.1	1.5	0.1	R20	5.1	3.8	1.5	0.3

Table 3 Nearest data for each data

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
18	13	4	3	1	19	3	1	4	2	20	8	2	9	17	15	11	1	6	5

Fig. 5 **a** Link structure between the data in the dataset. **b** Initial clusters based similarity

3.3.2 Clustering-based genetic algorithm

As the clustering implemented in the first phase has been on the basis of just similarities and relationships between the data, using a technique for combining such clusters with an appropriate measure and extracting the clusters with the most acceptable combination would be crucial. The technique applied in this research is the genetic algorithm.

As it is known, the input of the genetic algorithm is the average of data of each cluster in the first phase. For above

instance, according output of clustering-based similarity for 20 data in the first phase, the input genetic algorithm is a dataset with 4 data, each of them represents its cluster data. The respective benefits are the reduced complicatedness of the problem. Moreover, the data would not lose their clusters in the cluster integration procedure such that it would not break. Furthermore, implementation of the genetic algorithm is easier, and can be performed with less data.

Fig. 6 An example of a chromosome

3.3.2.1 Representation of the chromosomes Thus, it is necessary to represent the chromosomes in a way that the number of clusters may be concurrently determined by performing clustering of data at runtime. Hence, if we assume that the highest number of cluster centers equals K , and the number of dimensions or features of the dataset equals d , then each chromosome is a dimensional matrix $((K * (d + 1)))$. For obtaining optimum number of clusters, a decision variable would be added to each cluster center. This variable is an activation decision, which is a number between 0 and 1. If this value is ≥ 0.5 , then it is an active center; otherwise it would be disabled. Figure 6 depicts how each chromosome would be shown.

3.3.2.2 Fitness function The fitness function would be used to calculate the value of each chromosome in the genetic algorithm so that the distance between each sample and each cluster center has been firstly computed. Each sample has been allocated to a cluster with the smallest distance to the center of that cluster. This research assesses the final clusters and provided a novel evaluation criterion. A suitable clustering evaluation criterion follows both compactness and separation parameters. Therefore, the research created its evaluation function accordingly. It also called the fitness function as CSMC.

Therefore, for calculating the separation of a cluster from the other cluster, the distance between all components has been computed in association with another cluster. Equation (6) represents the separation of 2 clusters. Ultimately, total distance has been achieved as the separation of a cluster in association with another one. $|c_i|$ refers to the numbers of the components belonging to the cluster. $\vec{x}_{c_i}$ stands for the data that belong to the cluster c_i . K represents the number of active clusters in a chromosome, which would be obtained at the end. In fact, it is the number of active centers that is greater than 0.5.

Also, the smaller of the difference between the maximum distance and the minimum distance in each cluster indicates that the data in the cluster would have higher similarities.

Compactness of a cluster in Eq. (7) can be achieved from the difference in the sum of the highest distance and the sum of the lowest distance of each data.

$$Sep_{c_i, c_j} = \frac{1}{|c_i|} * \frac{1}{|c_j|} \sum_{\forall \vec{x}_i \in c_i, \vec{x}_j \in c_j} d(\vec{x}_i, \vec{x}_j) \quad (6)$$

$$Com_{c_i} = \frac{1}{2 * |c_i|^2} \left(\sum_{\forall \vec{x}_i \in c_i} \max(d_{\vec{x}_i}) - \sum_{\forall \vec{x}_i \in c_i} \min(d_{\vec{x}_i}) \right) \quad (7)$$

$$CSMC = \frac{1}{K * (K - 1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^K Sep_{c_i, c_j} - Com_{c_i} - Com_{c_j} \quad (8)$$

3.3.2.3 Primary population production It should be noted that a primary population should be existed for initiating the genetic algorithm. It is widely accepted that a random procedure is employed for producing primary population. A random technique possesses a high speed for inducing the primary population. Therefore, such a technique has been employed in this research.

3.3.2.4 Choosing parents According to this research design, in this phase, some parent chromosomes have been chosen on the basis of the degree of fitness they have taken from the evaluation function so that these chromosomes establish the offspring after applying the genetic operators, including crossover and mutation. Roulette wheel method is one of the most convenient algorithms that should be chosen. According to the above technique, each selection probability value has been firstly sorted together. Afterwards, a random number has been generated within the interval from 0 to 1. Therefore, the interval selection is based on the fact that the sum of the probability values consistently equals 1. Thus, the chromosome index corresponding to the random number has been determined via comparing the random number with the roulette wheel interval.

3.3.2.5 Crossover operator Then, the crossover operator combines chromosomes together for transferring the good genes of each chromosome to the child so that the general solution of the algorithm approaches the optimum solution for the problem. This research employed double-point crossover with a constant crossover probability of μ_c . This condition is described here with an example (Fig. 7).

3.3.2.6 Mutation operator The mutation operator attempts to move from one point to the other point in

Fig. 7 **a** Parent chromosome.
b Child chromosome using double point crossover

0.5	0.6	0.4	0.6	0.32
0.4	0.8	0.9	0.7	0.54
0.3	0.1	0.3	0.6	0.77
0.2	0.2	0.5	0.8	0.80
0.7	0.8	0.5	0.1	0.53

Parent 1

(a)

0.4	0.6	0.3	0.7	0.57
0.2	0.4	0.8	0.2	0.43
0.7	0.8	0.5	0.1	0.62
0.1	0.9	0.8	0.9	0.38
0.3	0.1	0.3	0.6	0.45

Parent 2

0.5	0.6	0.4	0.6	0.32
0.2	0.4	0.8	0.2	0.43
0.7	0.8	0.5	0.1	0.62
0.1	0.9	0.8	0.9	0.38
0.7	0.8	0.5	0.1	0.53

Child 1

(b)

0.4	0.6	0.3	0.7	0.57
0.4	0.8	0.9	0.7	0.54
0.3	0.1	0.3	0.6	0.77
0.2	0.2	0.5	0.8	0.80
0.3	0.1	0.3	0.4	0.45

Child 2

Fig. 8 Random mutation

0.5	0.6	0.4	0.6	0.32
0.4	0.8	0.9	0.7	0.54
0.3	0.1	0.3	0.6	0.77
0.2	0.2	0.5	0.8	0.80
0.7	0.8	0.5	0.1	0.53

Parent

→

0.5	0.6	0.4	0.6	0.32
0.4	0.8	0.9	0.7	0.54
0.3	0.1	0.5	0.6	0.77
0.2	0.2	0.5	0.8	0.80
0.7	0.8	0.5	0.1	0.53

Child

the search space via making modification in the chromosome genes. In the genetic algorithm, this operator may contribute significantly to escaping from the local optimum. This research applied a random mutation method. According to this technique, the one point has been initially chosen randomly. This data point is given within the allowed range of data. Figure 8 shows the random mutation.

3.3.2.7 Updating population After crossover and mutation operators have been applied, and the intended children have been created. The major question has been which of the chromosomes would be passed to the next generation for continuing the application of the algorithm on them. A method to update the population is to use an elite population. It should be noted that using elitism in the implementation of GA applies a significant impact on the finding of an optimum response and convergence of the algorithm. However, one of the existing techniques of elitism is to transmit N copies of the best chromosome or parent chromosomes to the new generation. The above technique enjoys a high efficiency [33]. In order to, analysis the ACGA algorithm, this approach is

Table 4 Result of the ACGA algorithm

Clustering technique	Number of cluster	Initial center of cluster	Execution time (s)	Accuracy (%)
ACGA	Automatic	Randomly	78	76.2

performed for the Wifi-localization dataset. The ACGA in first phase discover 455 cluster for dataset based similarity. In the second phase, number of optimal cluster is 4 based on GA algorithm. The experimentation is run for five times and Table 4 shows the results. Figure 9 presents the optimal clusters.

According to Table 4, ACGA has shown the best accuracy in comparison to GA based SSE, GA based DB, GenClust and ACDE algorithms. The run time of ACGA algorithm is lower than GenClust and ACDE, but it has higher execution time in comparison to GA based SSE and GA based DB techniques.

Fig. 9 Clustering of the Wifi-localization dataset by ACGA

3.4 The all k -th order Markov model for the clusters

It is widely accepted that displaying web transactions as the sequences of the viewing pages causes using a number of helpful models for discovering and analyzing the user's navigation patterns. One of such strategies is to model a user's survey behavior on the website by Markov chain. The order of Markov model is corresponding to the number of former events in anticipating an upcoming event. In particular, the Markov model of k order predicts the probability of the next event by watching the k event of the past. Correspondingly, when the user sessions have been clustered for discovering sequential patterns, the present research employed Markov model with all k -order for all clusters. Therefore, a matrix of the probability of the transfer has been created among web pages for a one-to- k order Markov model. Thus, for calculating the probabilities, if $P = \{p_1, p_{21}, p_3, \dots, p_m\}$ is a set of the pages in a website, the probability of $w_{\{j_1, j_2, \dots, j_k\} \rightarrow i}$ of a dataset would be approximated by Eq. (9):

$$\text{prob}(w_{\{j_1, j_2, \dots, j_k\} \rightarrow i}) = \frac{w_{\{j_1, j_2, \dots, j_k\} \rightarrow i}}{\sum_{p_m \in P} w_{\{j_1, j_2, \dots, j_k\} \rightarrow p_m}} \quad (9)$$

In Eq. (9), $1 \leq j \leq k$ where k stands for an equal order of Markov model. $w_{\{j_1, j_2, \dots, j_k\} \rightarrow i}$ refers to the numbers of the times page i appear after set $\{j_1, j_2, \dots, j_k\}$ across all sessions.

3.5 Suggestion of pages based on user behavior

After training and entering the current user session, for predicting in the online phase, it is initially necessary specify to which cluster the current session belongs, and then find the closest cluster to the current user session. Therefore, the final page of the user's current session in the suggested system has been eliminated and all sequences of 1 to k elements have been achieved on the basis of the last k pages of the session. Afterwards, the probability of visiting every page of the cluster has been determined, and the sequences of the current session have been derived from the transfer probability matrices created in the offline phase. Therefore, the probability of visiting the pages after sequences of one element from the 1st order matrix, the probability of visiting the pages after sequences of two elements from the 2nd order matrix, and the probability of visiting the pages after sequences of k elements have been taken out from the k -th order matrix. Next, the probabilities have been summed up in the sequences for similar number of pages as expressed in Eq. (10):

$$p(a_{n+1}|a_n a_{n-1} \dots a_{n-k}) = [p(a_{n+1}|a_n) + p(a_{n+1}|a_n a_{n-1}) + \dots + p(a_{n+1}|a_n a_{n-1} \dots a_{n-k})] / |N_p| \quad (10)$$

In Eq. (10), $|N_p|$ represents the number of probability greater than zero. This probability has been computed for all pages. Afterwards, the most probable pages have been recommended to the user from the pages that would likely

Table 5 The probability of pages after one to three sequences

P5	P1(0.25),P2(0.5),P4(0.25)	P1(0.25/1=0.25)
P3 P5	P4(1)	P2(0.5/1=0.5)
P4 P3 P5	P4(1)	P4(2.25/3=0.75)

to be viewed after the current session. For instance, if an active user possessed a session as shown below, Table 5 presents the probability of pages after one-item to three-item sequences:

$$User_{active} = P1, P2, P4, P3, P5, ?$$

Hence, page P4 with a probability of 0.75 would be offered to the user as the next page.

4 Time complexity analyses

This section deals with the timing of the suggested algorithms. Thus, the introduced recommender system, ACGA, and the modified all k-th MM have been analyzed. The notations employed in this section are defined as follows:

- S refers to the number of sessions.
- $MaxIteration$ denotes the highest number of iteration that the GA algorithm repeats.
- $NPOP$ stands for the number of population in GA algorithm.
- N_f is the number of final clusters.
- N_T represents the number of the testing data that recommender system recommends.
- k indicates the number of orders.
- N_p refers to the number of pages.
- N_R stands for the number of pages recommended to a user.

Lemma 1 The time complexity of ACGA algorithm is $O(S + MaxIteration * NPOP)$.

Proof In the first part of ACGA, the clustering-based similarity algorithm complexity largely is dependent on the number of sessions, $O(S)$. In the second part, the GA algorithm depends on the number of $MaxIteration$ and $NPOP$. Within a given iteration, a solution would be provided, the parameters should be filled for $NPOP$ population. This algorithm implemented $MaxIteration$ number of times. Hence, the complexity of ACGA algorithm is $O(S + MaxIteration * NPOP)$. $\square$

Lemma 2 The time complexity of modified all k-th order MM algorithm is $O(k * N_f * S)$.

Proof The running time of the modified all k-th order MM is linear because making each order of MM takes linear time with the number of session. Thus, to make all k-th order MM need $O(k * S) = O(S)$. This algorithm is implemented for each clustering that is found with ACGA algorithm. Moreover, the time complexity of this algorithm depends on N_f . Hence, the complexity of modified all k-th order MM is $O(k * N_f * S)$. $\square$

Lemma 3 The time complexity of the suggested recommender system is $O(ACGA) + O(modifiedallk - thorderMM) + O(N_p * N_R * N_T)$.

Proof Prior to implementing the major recommendation procedure, it is necessary that the clustering sessions create probability matrix for each cluster. With regard to the clustering algorithm and Markov procedure timing, complexity would be summed with each other in order to analyze the recommender system's timing. During the online phase, an active session should initially identify the best cluster. The clustering should make a comparison between 2 vectors of an active session and a centroid cluster. Hence, this comparison would depend on the number of pages in the system, with time complexity of $O(N_p)$. After determining the cluster of an active session, the high probability pages must be recommended to user. This phase would provide the timing analysis as $O(N_p * N_R)$. If the recommendation is iterated for N_T sessions, then the time complexity would be $O(N_p * N_R * N_T)$. $\square$

5 Evaluation and simulation results

Excel 2010 and MATLAB R2013a software have been applied for simulating the introduced system. The present research initially trained it via a training dataset. When the system has been trained, a set of test data have been employed that have not been contributed to the development of the sequential model for evaluating the suggested system and simulating the active user.

In order to achieve the above condition, each input vector in the test section, which showed the user interest, has been divided into 2 sections. Then, the first part of each vector has been investigated as the current user session, and the second part as the pages to be offered to the user. Afterwards, the output of the suggested system has been compared to the second section, and assessed the system function via attaining the precise number of pages introduced. Experimental analyses have been conducted on the basis of the suggested recommender system as provided in Algorithm 1.

Algorithm 1 Proposed RS

Input: Web users' sessions

Output: Recommendation list of pages

Begin:

Step 1: Making session vectors

1: **for** each session $s_i \in S$ **do**

2: Compute users' interest by using Equation (1), Equation (2), and Equation (3);

3: **end for**

Step 2: Web session clustering by using algorithm ACGA

//clustering-based similarity

4: **for** each session $s_i, s_j \in S$ **do**

5: create Similarity Matrix by using Equation (4) and Equation (5);

6: cluster each session with other session with the highest similarity in the Similarity Matrix

4: **end for**

// clustering-based GA

7: get result of clustering-based similarity as input

8: compute average of the session of each cluster

9: initialize population (select K clusters centroids randomly with activation threshold as Genomes)

10: evaluate the fitness function (CSMC) or cost function (CS) for each chromosome i in the population

11: **for** It=1 to MaxIteration **do**

12: perform the selection process based on Roulette Wheel

13: generate new chromosome double point crossover and random mutation

14: evaluate the fitness function (CSMC) or cost function (CS) for new chromosome

15: update the population for the next generation

16: **end for**

Step 3: Create probability Matrix for 1-to-k order MM

17: get N_c as number of final cluster

18: **for** order=1 to k **do**

19: **for** C=1 to N_c **do**

20: Compute probability by using Equation (9);

21: **end for**

22: **end for**

Step 4: Suggestion list of pages for given current user session x (online phase)

23: find closet cluster

24: used modified all k-th order MM based on Equation (10) for prediction pages;

25: recommendation pages with highest probability to current user.

5.1 The CTI dataset

This research applied the CTI² web records file data for implementing the suggested method. The DePaul CTI Web server has been used to collect the CTI dataset. This server is based on a random sampling of the users visiting the site for a 2-week interval during April 2002. The unfiltered data has been composed of 20,950 sessions from 5446 users. Put differently, the filtered data consisted of

13,745 sessions and 683 page views. In random, 9000 of the session in the dataset has been chosen to train, and another 4745 has been employed to test. Table 6 reports this data after its preprocessing operation.

Table 6 Dataset used in the experiment

Dataset	Period	Number of sessions	Number of pages
CTI	Two weeks period during April 2002	13,745	683

² facweb.cs.depaul.edu/mobasher/classes/ect584/resource.html.

Table 7 The parameters value for ACGA and ACDE

ACGA		ACDE	
Parameter	Value	Parameter	Value
Population size	30	Population size	10*dimension of dataset
The number of iteration	50	The number of iteration	30
Crossover probability (μ_c)	0.20	Crossover rate (CR_{max})	1
Mutation probability μ_m	0.15	Crossover rate (CR_{min})	0.5
K_{min}	2	K_{min}	2
K_{max}	20	K_{max}	20

5.2 Effective parameters in evaluation

Assessment of the clustering is one of the most significant phases for analyzing the clustering. The nature of the unsupervised clustering has transformed clustering evaluation procedure into one of the most robust clustering analysis phases. Validation techniques for clustering are grouped into 2 major kinds of internal clustering validation and external clustering validation [13]. The CS and new validation measure, which has been provided by this study researcher, have been used as a validation procedure to calculate the inside and outside of the cluster based on the pairwise distance of the cluster objects. CS is defined as Eq. (12); where N_i stands for the number of the components in the i th cluster C_i . Prior to the implementation of the CS measure, the centroid of a cluster has been calculated through averaging the data vectors that belongs to the cluster via Eq. (11):

$$\vec{m}_i = \frac{1}{N_i} \sum_{\vec{x}_j \in C_i} \vec{x}_j \quad (11)$$

$$CS = \frac{\sum_{i=1}^K \left[\frac{1}{N_i} \sum_{\vec{x}_i \in C_i, \vec{x}_q \in C_i} \max \{ d(\vec{x}_i, \vec{x}_q) \} \right]}{\sum_{i=1}^K \left[\min_{j \in K, j \neq i} \{ d(\vec{m}_i, \vec{m}_j) \} \right]} \quad (12)$$

The low CS and high CSMC measures in the clustering algorithm revealed that the clustering could maintain acceptable compression and separation concurrently.

The numbers of the suggested pages and 3 criteria of coverage, accuracy, and F-measure are the factors affecting the system performance. According to the studies, accuracy represents the capability of the recommender system for generating exact suggestions. Put differently, the accuracy of the offer is equal to the ratio of the precise suggestions to the total suggestions.

$$accuracy(s, r) = \frac{|s \cap r|}{|r|} \quad (13)$$

It is widely accepted that the coverage refers to the capability of the recommender system for generating each offer viewed by the user. Actually, the coverage equals the proportion of the well-received suggestions to the entire remaining pages of the same session.

$$coverage(s, r) = \frac{|s \cap r|}{|s|} \quad (14)$$

The F-measure is a combination of two coverage and accuracy metrics in a single metric:

$$F - measure = 2 * \frac{accuracy(s, r) * coverage(s, r)}{accuracy(s, r) + coverage(s, r)} \quad (15)$$

5.3 Results of clustering and evaluation of suggestion pages

When the user sessions have been identified by the linear function in Eq. (3), the vector for each session has been established based on the parameters of $\lambda = 0.6$. When the users' interests in the pages have been revealed and the vectors have been created, the user sessions have been clustered via the ACGA algorithm. The part 1 of ACGA finds 1243 cluster-based similarity and the mean of each cluster sent to GA in part 2 for discovering the final clusters. The outputs of ACGA algorithm have been compared with K-means and ACDE with regard to the CS and CSMC measure. Table 7 presents the values of the parameters of the genetic algorithm in ACGA and ACDE. For ACDE [29], the

Table 8 Final solution for each algorithm with CS and CSMC measures

Algorithm	Avg no. of clusters found	CS measure	Avg no. of clusters found	CSMC measure
K-means	11.00	2.1333	6.00	0.3202
ACDE	2.4	1.6223	3.00	0.3764
ACGA	8.8	1.5126	6.00	0.4029

Table 9 Experiments on the effect of number of clusters (K) in K-means algorithm based CS and CSMC measures

Number of cluster	CS measure	CSMS measure	Number of cluster	CS measure	CSMS measure
K=2	2.8421	0.2979	K=17	2.5892	0.2897
K=3	3.1079	0.2927	K=18	2.5026	0.3058
K=4	2.8392	0.2916	K=19	2.4960	0.3022
K=5	2.9024	0.3000	K=20	2.2442	0.2980
K=6	2.8468	0.3202	K=21	2.2653	0.3015
K=7	2.7692	0.2843	K=22	2.3247	0.3008
K=8	2.6648	0.2849	K=23	2.4944	0.2983
K=9	2.3685	0.2842	K=24	2.4453	0.3126
K=10	2.8997	0.3070	K=25	2.3090	0.3174
K=11	2.1333	0.3031	K=26	2.3642	0.3136
K=12	2.5040	0.2959	K=27	2.2449	0.3050
K=13	2.5520	0.3047	K=28	2.1834	0.3160
K=14	2.6190	0.2966	K=29	2.3111	0.2989
K=15	2.2384	0.3201	K=30	2.2711	0.2972
K=16	2.5795	0.3099	Min(CS)=2.1333	Max(CSMC)=0.3202	

Since the lowest value for CS and the highest value for CSMC measures are best state, this is shown in bold

Table 10 CS measure of ACGA for 10, 30, and 50 population size $|P_s|$

$ P_s $	ACGA ($ P_s = 10$)	ACGA ($ P_s = 30$)	ACGA ($ P_s = 50$)
CS measure	2.1891	1.5126	1.2972
Avg no. of clusters found	10.2	8.8	9.6

Table 12 CS measure of ACGA for 20, 30, and 40 generations

N	ACGA (N = 20)	ACGA (N = 30)	ACGA (N = 40)
CS measure	1.5563	1.5126	1.5208
Avg no. of clusters found	7.6	8.8	10.00

present research used the parameters recommended in their study. Moreover, Table 8 shows the results of the 3 algorithms.

As K-means algorithm gains the number of the clusters from user, the present research experimented this algorithm with some distinct clusters using 2 measures for evaluating the clusters. Table 9 reports the results for different Ks. This algorithm outputs are indicated by the number of various clusters. When the K-means algorithm uses CS measure, the minimum value with 11 clusters would be obtained where the best state for clustering the data is 11. Nevertheless, if CSMC is employed to assess the cluster, the best state for clustering would be 6, in which the highest CSMC equals 0.3202.

The major objective has been examining if the final clusters achieved by ACGA are more acceptable than other clustering methods with regard to the fitness function

(CSMC) and cost function (CS) or not. Outputs revealed the average number of clusters and measures after five independent runs in 3 methods. It has been found that ACGA possessed less CS and greater CSMC values in various runs.

ACGA applies some factors such as total number of generations, $|P_s|$, and number of iterations. Default values employed for the above factors in this research experiments include number of generations ($N = 30$), $|P_s| = 30$, number of iterations ($iter = 50$). The research studied the effects of the factors on the ACGA performance. It also used 3 distinct population sizes ($|P_s| = 10, 30$, and 50) rather than only the default $|P_s| = 30$. Tables 10 and 11 present the average CS and CSMC measure and the number of clusters of five runs of ACGA with 3 distinct $|P_s|$ values. As seen, the clustering outputs usually ameliorate with an enhancement in the population size.

Table 11 CSMC measure of ACGA for 10, 30, and 50 population size $|P_s|$

$ P_s $	ACGA ($ P_s = 10$)	ACGA ($ P_s = 30$)	ACGA ($ P_s = 50$)
CSMC measure	0.3580	0.4029	0.4300
Avg no. of clusters found	9.2	6.00	8.00

Table 13 CSMC measure of ACGA for 20, 30, and 40 generations

<i>N</i>	ACGA (<i>N</i> = 20)	ACGA (<i>N</i> = 30)	ACGA (<i>N</i> = 40)
CSMC measure	0.4046	0.4029	0.4095
Avg no. of clusters found	7.8	6.00	7.00

The present research applied 3 distinct numbers of generation (20, 30, and 40) for ACGA instead of only the default 30 generations. Tables 12 and 13 report the average CS and CSMC measure values of five runs of ACGA with 3 distinct numbers of generations. As shown in the tables, a greater numbers of generations usually provide more

Table 14 CS measure of ACGA for 10, 50, and 100 iterations

<i>Iter</i>	ACGA (<i>iter</i> = 10)	ACGA (<i>iter</i> = 50)	ACGA (<i>iter</i> = 100)
CS measure	1.8542	1.5126	1.3218
Avg no. of clusters found	10.2	8.8	7.2

acceptable clustering outputs for CSMC measure; however, the default generation would be better in CS measure.

Moreover, the research utilized 3 various numbers of iterations (*iter* = 10, 50, 100) for the ACGA instead of the default 50 iterations. According to Tables 14 and 15, ACGA with 100 iterations would be accompanied by the best clustering; a result, which has been predicted earlier.

As the clustering of the users has been ended, if the initial number of chromosomes is 50, the algorithm would have more acceptable results. The present research employed cluster outputs through CS and CSMC measure in order 9 and 8 cluster, and Markov model of order 1-3 has been used to each of the clusters and its probability matrix has been specified. When new users have been entered, and the introduced recommender system has been implemented,

Table 15 CSMC measure of ACGA for 10, 50, and 100 iterations

<i>Iter</i>	ACGA (<i>iter</i> = 10)	ACGA (<i>iter</i> = 50)	ACGA (<i>iter</i> = 100)
CSMC measure	0.3558	0.4029	0.4057
Avg no. of clusters found	9.8	6.00	7.00

Table 16 Results of different Markov models in page suggestions with 9 clusters in CS measure

Number of page recommendation		1	2	3	4	5	6	7	8	9	10	11
1st-MM	Accuracy	0.41	0.31	0.28	0.24	0.22	0.20	0.20	0.19	0.17	0.18	0.18
	Coverage	0.98	0.98	0.99	0.98	0.99	0.99	0.99	0.99	0.99	1.00	0.99
2nd-MM	Accuracy	0.36	0.29	0.27	0.26	0.24	0.23	0.21	0.21	0.19	0.19	0.19
	Coverage	0.80	0.82	0.82	0.80	0.79	0.79	0.78	0.74	0.75	0.72	0.73
3rd-MM	Accuracy	0.34	0.28	0.27	0.25	0.23	0.23	0.20	0.22	0.20	0.19	0.18
	Coverage	0.57	0.55	0.51	0.48	0.47	0.48	0.44	0.44	0.41	0.41	0.39
New All-3th-order MM	Accuracy	0.44	0.32	0.29	0.26	0.23	0.22	0.22	0.20	0.18	0.19	0.18
	Coverage	0.98	0.98	0.99	0.98	0.99	0.99	0.99	0.99	0.99	1.00	0.99

Table 17 Results of different Markov models in page suggestions with 8 clusters in CSMC measure

Number of page recommendation		1	2	3	4	5	6	7	8	9	10	11
1st-MM	Accuracy	0.43	0.32	0.30	0.26	0.24	0.22	0.21	0.19	0.18	0.18	0.18
	Coverage	0.98	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.99	1.00	0.98
2nd-MM	Accuracy	0.39	0.32	0.29	0.27	0.26	0.24	0.22	0.20	0.20	0.19	0.19
	Coverage	0.81	0.83	0.82	0.79	0.78	0.77	0.77	0.74	0.73	0.70	0.71
3rd-MM	Accuracy	0.35	0.29	0.27	0.26	0.25	0.23	0.22	0.22	0.19	0.19	0.18
	Coverage	0.57	0.56	0.50	0.47	0.46	0.47	0.43	0.43	0.39	0.40	0.39
New all-3th-order MM	Accuracy	0.45	0.34	0.30	0.28	0.25	0.24	0.23	0.21	0.21	0.19	0.19
	Coverage	0.98	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.99	1.00	0.98

Fig. 10 Comparison of the proposed algorithm with other algorithms in terms of accuracy measure

Fig. 11 Comparison of the proposed algorithm with other algorithms in terms of coverage measure

Fig. 12 Comparison of the proposed algorithm with other algorithms in terms of F-measure measure

Fig. 13 Computational time of the techniques for CTI data set

the outputs of the suggested pages have been computed on the basis of the accuracy and coverage factors with the use of Markov models of order 1–3 and the Markov model with all three new orders in Tables 16 and 17.

It has been demonstrated that as the recommender system employs novel Markov model and CSMC for fitness function in ACGA, it would have higher accuracy and coverage in comparison with other models of Markov. A reason for this condition may be that the system applies

the 3 orders of the Markov model under a single form and covers all three, which causes the probability of the identical pages looks even more significant at any order of Markov's model. However, its recommender system is particularly compared with various methods, including HSCR [8], HKSCR [8], IHKSCR [8], KMM, and KMMPSPR [30]. Findings revealed that its suggested systems had more acceptable performance. Figures 10, 11, and 12 show the outputs.

Notably, evaluation of the suggestions strongly depend on the number of the pages that must be presented to the current users, and each set of pages possesses distinct coverage and accuracy, which are associated with the number of pages. According to the result, the proposed technique enhances a set of suggested pages, and exactly identifies the users' interests while keeping great accuracy. Inserting 2 factors of accuracy and coverage in the form of the F-measure harmonic function represents the importance of the proposed algorithm in comparison with other algorithms.

This paper presents the results on execution time required by techniques in Fig. 13. The complexity of HSCR is $O(T_h * S) + O(N_p * N_R * N_T)$ [8], where T_h the number of iteration is that harmonic search repeats. The complexity of HKSCR is $O(T_{hs} * T_{KM} * d * K * S^2) + O(N_p * N_R * N_T)$ [8], where T_{KM} , d , and K is the number of iteration is that K-means algorithm iterates, the number of dimensions and number of clusters.

Also the complexity of IHKSCR is $O(g * T_{hs} * T_{KM} * d * K * S^2) + O(N_p * N_R * N_T)$, where g is the number of generations, that recommender system proposes recommendation [8].

The complexity of the KMMPSPR algorithm is $(T_{KM} * d * K * S) + O(k * K * S) + O(P * k * K * S) + O(N_p * N_R * N_T)$ [30], P is the number of page in data set. The complexity of KMM algorithm is complexity of $O(T_{KM} * d * K * S) + O(k * K * S) + O(N_p * N_R * N_T)$.

The results indicate that the proposed algorithm is computationally more expensive than the KMMPSPR [30] and KMM algorithms, but it is better than HSCR [8], HKSCR [8] and IHKSCR [8].

6 Conclusion

The present research proposed a recommender system for offering web page suggestions to the web users. Particularly, the user sessions in the first phase have been vectored on the basis of the user interest. Moreover, an automatic clustering algorithm has been used to cluster them. In addition, the modified Markov models from order 1 to 3 have been applied for building the probability matrix. Furthermore, during the online phase, the recommender system identified the nearest cluster as soon as the user logged in, and then proposed new pages through the Markov model with all 3th order. The suggested system has been eventually contrasted to the basic algorithm for coverage, accuracy, and F-measure. It has been found that the system possessed a greater accuracy based on the set of the introduced pages. There are two limitations against the advantages of the proposed recommender system. Creating a probability matrix in Markov models expands a big memory. Some columns in this matrix have zero

probability and didn't used. A possible solution is to use a hash table to store non-zero probability data. Another limitation is the random selection of the initial centers of the clusters in the second phase of ACGA algorithm. To increase the accuracy of clustering, a possible solution can be the selection of an initial centers based on the deterministic and random process. These problems will be improved in the future.

Acknowledgements The researcher appreciates Dr Ali Harounabadi and Fateme Hamzeheian for helping in discussion of this research. I wish to sincerely thank to the honored reviewers for their comments that helped me for improving the research.

Compliance with ethical standards

Conflict of interest The corresponding author states that there is no conflict of interest.

References

1. Almutadha Y, Bin Sulaiman M, Mustapha N, Udzir N (2011) IPACT: improved web page recommendation system using profile aggregation based on clustering of transactions. *Am J Appl Sci* 8(3):277–283
2. Alyari F, Jafari Navimipour N (2018) Recommender systems: a systematic review of the state of the art literature and suggestions for future research. *Kybernetes* 47(5):985–1017
3. Chen J, Wang B, Liji U, Ouyang Z (2019) Personal recommender system based on user interest community in social network model. *Phys A* 526:1–14
4. Chou C, Su M, Lai E (2004) A new cluster validity measure and its application to image compression. *Pattern Anal Appl* 7(2):205–220
5. Ciobanu D, Dinuca CE (2012) Predicting the next page that will be visited by a web surfer using page rank algorithm. *Int J Comput Commun* 1(6):60–67
6. Deshpande M, Karypis G (2004) Selective markov models for predicting web page accesses. *ACM Trans Internet Technol (TOIT)*. 4(2):163–184
7. Dhyani D, Bhowmick SS, Ng W (2003) Modelling and predicting web page accesses using markov processes. In: 14th international workshop IEEE database and expert systems applications, pp 332–336
8. Forsati R, Moayedikia A, Shamsfard M (2015) An effective web page recommender using binary data clustering. Springer, New York, pp 1–48
9. Hu M, Liu B (2004) Mining and summarizing reviews. In: Proc of ACM SIGKDD Intl on Knowledge Discovery and Data Mining (KDD'04), pp 168–172
10. Isinkaye F, Folajimi Y, Ojokoh B (2015) Recommendation systems: principles, methods and evaluation. *Egypt Inform J* 16:261–273
11. Lin H, Yang F, Kao Y (2005) An efficient GA-based clustering technique. *Tamkang J Sci Eng* 8(2):113–122
12. Liu H, Keşelj V (2007) Combined mining of web server logs and web contents for classifying user navigation patterns and predicting user's future requests. *Data Know Eng* 61:304–330
13. Liu Y, Li Z, Xiong H, Gao X, Wu S (2013) Understanding and enhancement of internal clustering validation measures. *IEEE Trans Cybern* 43(3):982–993

14. Lloyd SP (1992) Least squares quantization in PCM. *IEEE Trans Inf Theory* 28(2):129–136
15. Maulik U, Bandyopadhyay S (2000) Genetic algorithms-based clustering technique. *Pattern Recogn* 33:1455–1465
16. Mamoun A, Khalil I (2012) Prediction of user's web-browsing behavior: application of Markov model. *IEEE Trans Syst Man Cybern Part B Cybern* 42(2):1131–1142
17. Mishra R, Kumar P, Bhasker B (2015) A web recommendation system considering sequential information. *Decis Support Syst* 75:1–10
18. Milovančević NS, Gračanac A (2019) Time and ontology for resource recommendation system. *Phys A* 525:752–760
19. Modarresi K (2016) Recommendation system based on complete personalization. *Proc Comput Sci* 80:2190–2204
20. Mulyawan B, Christani V, Wenas R (2019) Recommendation product based on customer categoration with K-means clustering method. In: *IOP conference series materials science and engineering*, vol 508, pp 1–7
21. Narvekar M, Banu SS (2015) Predicting user's web navigation behavior using hybrid approach. *Int Conf Adv Comput Technol Appl (ICACTA)* 45:3–12
22. Pitkow J, Pirplli P (1999) Mining longest repeating subsequence to predict World Wide Web surfing. In: *2nd USENIX symposium on internet technologies and systems*, Boulder, CO
23. Poornalatha G, Raghavendra RS (2011) Web user session clustering using modified K-means algorithm. In: *Communications in computer and information science (CCIS), Advances in computing and communications (ACC 2011)*, vol 191, pp 243–252
24. Poonalatha G, Prakash SR (2017) Session based collaborative filtering for web page recommender (SCFR) system based on clustering. *Int J Control Theory Appl (IJCTA)* 10(8):679–687
25. Kumar P, Kumar V, Thakur RS (2019) A new approach for rating prediction system using collaborative filtering. *Iran J Comput Sci* 2:81
26. Rahman A, Islam Z (2014) A hybrid clustering technique combining a novel genetic with K-means. *Knowl-Based Syst* 71:345–365
27. Selvi C, Sivansankar E (2017) A novel optimization algorithm for recommender system using modified fuzzy c-means clustering approach. *Soft Comput* 23(6):1901–1916
28. Singh M (2004) *The practical handbook of internet computing*, 1st edn. CRC Press, New York
29. Swagatam D, Ajith A, Amit K (2008) Automatic clustering using an improved differential evolution algorithm. *IEEE Trans Syst Man Cybern Part A Syst Hum* 38(1):218–237
30. Thew P (2014) Web page access prediction based on an integrated approach. *Int J Comput Sci Bus Inform* 12(1):55–64
31. Thiyagarajan R, Thangavel K, Rathipriya R (2014) Recommendation of web pages using weighted K-Means clustering. *Int J Comput Appl* 86(14):44–48
32. Vijaya Kumar T, Guruprasad HS (2015) Clustering of web usage data using hybrid K-means and PACT algorithm. *Int J Inf Technol* 7(2):871–876
33. Yang C, Thiphuong Quyen N (2018) Data analysis framework of sequential clustering and classification using non-dominated sorting genetic algorithm. *Applied Soft Computing*. Elsevier, New York, pp 1–15
34. Zheng N, Li Q (2011) A recommender system based on tag and time information for social tagging systems. *Expert Syst Appl* 38:4575–4587

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.