International Journal of Soft Computing 6 (5-6): 168-174, 2011
ISSN: 1816-9503

A New Evolutionary Coclustering Approach for Web User Profiling

R. Rathipriya and K. Thangavel
Department of Computer Science, Periyar University, Salem, Tamil Nadu, India

Abstract: Coclustering is a two-way clustering approach involving simultaneous clustering along two dimensions of the data matrix. Extraction of coclusters comprises of web objects (i.e., web users and web pages) is an emerging research topic in the context of web usage mining. It overcomes some of the problems associated with traditional clustering methods by allowing automatic discovery of browsing pattern based on a subset of attributes. A correlated cocluster of clickstream data is a local pattern such that users in cocluster exhibit correlated browsing pattern through a subset of pages of a web site. This study aims to provide a new correlated coclustering framework using genetic algorithm to identify overlapping correlated cocluster from clickstream data. Experiment is conducted on the benchmark dataset. Results demonstrate the efficiency and beneficial outcome of the proposed method by correlating the users and pages of a web site in high degree. It also outperforms the existing traditional clustering of web usage data.

Key words: Coclustering, correlated cocluster, clickstream data, genetic algorithm, greedy search procedure, web usage mining

INTRODUCTION

Web mining (Cooley *et al*., 1999; Mobasher *et al*., 2002) scovers and extracts interesting pattern or knowledge from web data. It is classified into three types as web content mining, web structure and web usage mining. Web usage mining is the intelligent data mining technique for automatic discovery of patterns in clickstreams and associated data collected or generated as a result of user interactions with one or more web sites. Discovered patterns are usually represented as collections of pages, objects or resources that are frequently accessed by groups of users with common interests. These patterns are analyzed to determine user's behaviour which is an important and challenging research area in the web usage mining. In literature, clustering (Lee and Fu, 2008) is the widely used data mining technique to discover pattern of group of users with similar interest and motivation for visiting the particular website can be found by clustering. It is used to cluster the web users or web pages based on the similarities exists among them.

Based on a general understanding of the user's behavior, the subsets of users are highly correlated under certain web pages of a web site. But clustering is based on the assumption that all the related users behave similarly across all the pages of a web site or vice versa. Therefore, traditional clustering fails to identify such clusters from web usage data such as clickstream data.

To overcome the problem of clustering, concept of Coclustering or Biclustering was introduced. Biclustering was first introduced by Hartigan (1972). Coclustering is also known as biclustering, bi-dimensional clustering and subspace clustering in the literature (Ben-Dor *et al*., 2003). Coclustering attempts to cluster web users and web pages simultaneously based on the users' behavior recorded in the form of clickstream data called coclusters. These coclusters are representing the correlated browsing patterns which play a vital role in e-Commerce based applications. The main goal of this study is to identify maximal subgroups of users and pages such that these users and pages are correlated highly.

Recommender systems analyze patterns of user browsing interest and to provide personalized services which match user's interest in most business domains, benefiting both the user and the merchant. Task of these systems is to filter information by identifying the relevance of an item such as a browsing pattern of a given user. Collaborative filtering techniques are used to analyze the relations between similar users or similarity among pages to identify neighborhoods of like-minded users with similar browsing interest/behavior or of similar pages. For the above two applications, coclustering approach is the most appropriate method to analyze the user's browsing pattern along the both dimensions.

RELATED WORK

Web usage clustering (Rana *et al*., 2010; Srivastava *et al*., 2000) is a well studied problem and numerous clustering techniques are available in the literature. Web usage clustering can be either user or page clustering based on the information recorded in the web

Corresponding Author: R. Rathipriya, Department of Computer Science, Periyar University, 636011 Salem, India

server log file. Page clustering approaches based on the information extracted from page content, structure and also usage data. It aims to discover web page groups sharing similar functionality or semantics.

Web user clustering techniques can be used to discover users browsing behavioral patterns and associations between the web users and pages from the perspective of web user. Each user session could be expressed by a pageview vector with weights and the clustering on user sessions leads to extraction of the aggregate user sessions which could be viewed as user access patterns. Mobasher *et al.* (2002) employed the traditional K-means clustering algorithm to characterize user access patterns for web personalization based on mining web usage data. Xu *et al.* (2005) used Probabilistic Semantic Latent Analysis (PLSA) model to discover user access patterns and web page segments from web log files.

The clustering algorithms described above are mainly manipulated on one dimension (i.e., either users or pages) of the web usage data only rather than taking in to account the correlation between web users and pages. However, in most cases the web clusters do often exist in the forms of co-occurrence of pages and users i.e., users from the same group are particularly interested in one subset of web pages. Dhillon (2001) and Dhillon *et al.* (2003) proposed an innovative coclustering algorithm for text mining that monotonically increases the preserved mutual information by intertwining both the row and column clustering at all stages and it has been widely utilized in various fields such as social tagging system, target group identification and gene expression analysis (Ben-Dor *et al*., 2003; Cho *et al*., 2004; Kluger *et al*., 2003) etc.

Guandong *et al.* (2010) presented an algorithm using bipartite spectral clustering to cocluster web users and pages and the impact of using various clustering algorithms is also investigated in that study. Koutsonikola and Vakali (2003), proposed a fuzzy biclustering approach for web data which identifies groups of related web users and pages using spectral clustering method on both row and column dimensions (Table 1).

Most of the above mentioned works used Mean Squared Residue (MSR) score (Bagyamani and Thangavel, 2010) as merit function for extracting the biclusters with correlated pattern but interesting and relevant patterns such as shifting and scaling browsing patterns may not be detected using this measure (Bagyamani and Thangavel, 2010). However, it is important to discover this type of patterns since commonly the users can have a similar behavior although their frequency of visit varies in different ranges or magnitudes. This study presents a coclustering framework using genetic algorithm with the aim of finding optimal coclusters from web usage data. In this research, the proposed fitness function is based on the average correlation value among users or pages of web usage data to detect shifting and scaling patterns from it.

Table 1: Coclustering algorithms for microarray data analysis

Biclustering algorithms	Description
Cho *et al.* (2004)	K-means based biclustering algorithms was proposed that identifies m row clusters and n column clusters simultaneously while monotonically decreasing the mean square residue defined by Cheng and Church
Tang and Zhang (2001)	Interrelated Two Way Clustering (ITWC) algorithm was proposed that combines the results of one way clustering on both dimensions of the data matrix in order to produce biclusters
Busygin *et al.* (2002)	Double Conjugated Clustering (DCC) was proposed which aims to discover biclusters with coherent values defined using multiplicative model of bicluster
Getz *et al.* (2000)	Coupled Two Way Clustering algorithm was introduced which performs one way clustering on the rows and columns of the data matrix using stable clusters of row as attributes for column clustering and vice versa
Xie *et al.* (2007)	Hybrid evolutionary optimization algorithm based on particle swarm and Genetic algorithms was presented in this study to solve the biclustering problem
Chakraborty and Maka	In this study, biclustering algorithm using (2005) genetic algorithm was proposed that embeds greedy algorithm as local search procedure to find the best biclusters

METHODS AND MATERIAL FOR COCLUSTERING FRAMEWORK

Clickstream data: Clickstream data is cleaned and partitioned into a set of user transactions representing the activities of each user during different visits to the site. This data needs to be transformed and aggregated in to sessions. A session is a sequence of pageviews by a single user during a single visit $s = \{p_1, p_2, p_3, \ldots, p_n\}$. Let A (U, P) be a session-pageview matrix of size n x m where n is the number of sessions and m is the number of pageviews. Each row represents a session and each column represents a frequency of occurrence of the page view in the session. It is used to describe the relationship between web pages and users who access these web pages. The element a_{ij} of A (U, P) represents frequency of the user U_i of U visit the page P_j of P during a given period of time:

$$a_{ij} = \begin{cases} \text{Hits}(U_i, P_j) \text{ if } P_j \text{ is visited by } U_i \\ 0, \text{ otherwise} \end{cases} \quad (1)$$

Where:

Hits (U_i, P_j) = The count/frequency of the user
U_i = Accesses the page
P_j = During a given period of time

Coclustering approach: Unlike clustering, coclustering identifies groups of users that show similar browsing patterns under a specific subset of the pages of a web site called cocluster.

Coclusters preserve locally defined degree of homogeneity between users and pages of web site. Coclustering is the key technique to use when:

- Only a small set of the users visits the subset set of pages
- An interesting users' browsing pattern is exhibited only in a subset of the pages

Average Correlation Value (ACV): It is used to evaluate the homogeneity of a cocluster. Matrix B = (b_{ij}) has the ACV which is defined by the following function:

$$\overline{R(B)} = \max\left\{\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left|r\_row_{ij}\right| - n}{n^2 - n}, \frac{\sum_{k=1}^{m}\sum_{l=1}^{m}\left|r\_col_{kl}\right| - m}{m^2 - m}\right\} \quad (2)$$

r_row_{ij} = The correlation between row i and row j

r_col_{kl} = The correlation between column k and column l

ACV can tolerate translation as well as scaling. It works well for coclusters in which there's a linear correlation among the users or pages.

Cocluster with perfect shifting and scaling patterns has an average correlation value of 1 while that the cocluster without patterns has an average correlation value close to 0.

Figure 1 shows the patterns of lowly and highly correlated users of the biclusters that have ACV 0.3 and 1, respectively.

Clustering using Genetic Algorithm (GA)

Fitness function (F-I): The standard K-means algorithm using GA (Rathipriya *et al*., 2011; Sujatha and Iyakutty, 2010) which minimizes the intra cluster distance. The fitness value of population is computed by the following fitness function:

Objective function $F1 = \Sigma\| U^i - \mu^j \|$, i = 1… n and j = 1..k

Where:

n = The number of users

k = The number of clusters respectively

U^i = The user i

μ^j = The cluster centre

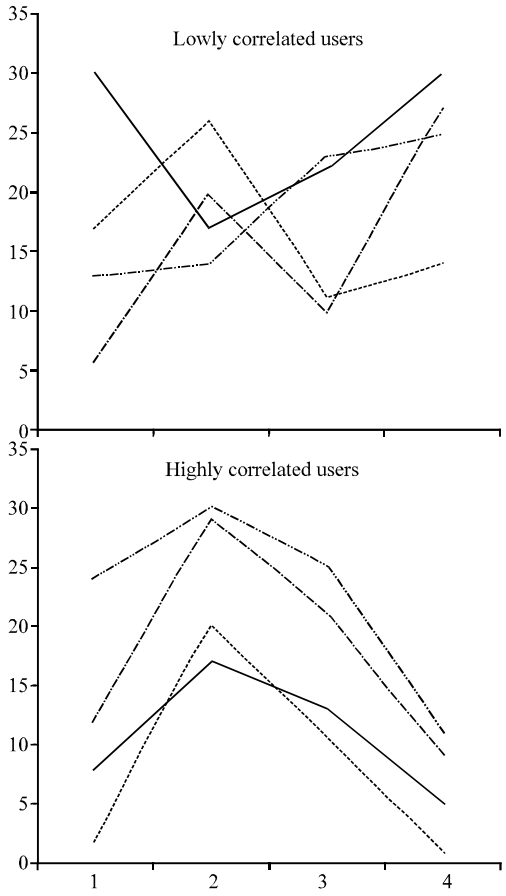

Fig. 1: Correlated users in the cocluster

Fitness function (F-II): The standard K-means algorithm using GA is designed and implemented to increase the average correlation value of the cluster by using the following fitness function F-II:

$$F2(u, p) = \begin{cases} |u| * |p|, & \text{if ACV (cluster) is} > \text{AVC threshold } \delta \\ 0, & \text{Otherwise} \end{cases} \quad (3)$$

where, |u| and |p| are number of users and pages in cluster and ACV threshold δ is predefined according to the problem.

Encoding of coclusters: Each cocluster is encoded as a binary string. The length of the string is the number of rows plus the number of columns of the user access matrix A (U, P). A bit is set to one when the corresponding user or page is included in the cocluster. These binary encoded coclusters are used as initial population for genetic algorithm.

Degree of overlapping: Degree of overlapping (Das *et al*., 2008) is used as quantitative index to evaluate

quantitatively the quality of generated coclusters. The degree of overlapping among all biclusters is defined as follows:

$$R = \frac{1}{|U|*|P|}\sum_{i=1}^{|U|}\sum_{j=1}^{|P|} T_{ij} \quad (4)$$

Where:

$$T_{ij} = \frac{1}{(N-1)}*\left(\sum_{K=1}^{N} W_k(a_{ij}) - 1\right)$$

Where:

N = The total number of coclusters
|U| = The represents the total number of users
|P| = The represents the total number of pages in the data matrix A

The value of $w_k(a_{ij})$ is either 0 or 1. If the element (point) a_{ij}, A is present in the kth cocluster then $w_k(a_{ij}) = 1$, otherwise 0. Hence, the R index represents the degree of overlapping among the coclusters. If R index value is higher, then degree of overlapping of the generated coclusters would be high. The range of R index is $0 \le R \le 1$.

COCLUSTERING FRAMEWORK FOR CLICKSTREAM DATA USING GENETIC ALGORITHM

In this research, coclusters with highly correlated users and high volume are preferred. Therefore, the fitness function based on ACV used to evaluate the quality of coclusters first step is to identify the initial coclusters called seeds by using K-means clustering algorithm. Second step is to enlarge and refine these seeds using greedy search procedure. But most of the times, it stuck at local optima. To escape from the local optima problem, genetic algorithm is used as optimization tool in the third step to extract global optimal coclusters from clickstream data. One-to-one relation between web users and pages of a web site is not appropriate because web users are not strictly interested in one category of web pages. Therefore, the proposed algorithm is tuned to discover the overlapping coclusters from clickstream data patterns. These overlapped coclusters have high degree of correlation among subset of users and subset of related pages of a web site.

Cocluster formation using K-means algorithm: In this study, K-means clustering method is applied on the session-pageview matrix A (U, P) along both dimensions separately to generate k_u user clusters and k_p page clusters. And then combine the results to obtain small co-regulated submatrices called coclusters.

Given a matrix A, let k_u be the number of clusters on user dimension and k_p be the number of clusters on page dimension after K-means clustering is applied. C^u is the set of user clusters and C^p is the set of page clusters. Let c_i^u be a subset of users and c_i^u, C^u $(1 \le i \le k_u)$. Let c_j^p be a subset of pages and c_j^p, C^p $(1 \le j \le k_p)$. The pair $(c_i^u; c_j^p)$ denotes a cocluster of A . By combining the results of user dimensional clustering and page dimensional clustering, k_u x k_p coclusters are obtained. These correlated coclusters are called seeds. The proposed coclustering framework inherits the simplicity, efficiency and wide applicability of the K-means algorithm.

Coclustering using greedy search procedure: A greedy algorithm repeatedly executes a search procedure which tries to maximize the ACV of the cocluster based on examining local conditions with the hope that the outcome will lead to a desired outcome for the global problem.

In this step initial seeds are enlarged and refined by adding/removing the rows and columns to/from the coclusters to extend their size and increase their homogeneity called ACV. Each seed is enlarged on both dimensions to increase its volume at same time ACV of the seed goes up incrementally. After seed enlargement, enlarged seeds are refined on the both dimensions to get optimal coclusters.

Algorithm 1
Coclustering using greedy search procedure:

Input: User access matrix A
Output: Set of enlarged and refined coclusters

```
Compute ku user clusters and kp page clusters from preprocessed
clickstream data
Combine ku and kp clusters to form ku x kp
coclusters called seeds
For each seed do
  Add/remove the element(user/page) to/from the cocluster
which increases the ACV of seed
End for
Return enlarged and refined coclusters
```

Enlarging and refining the seed starts from page list followed by user list until ACV is increased using greedy search procedure. In these algorithms, a user or page called node is added or removed to/from the cocluster. Either incremental value of ACV after adding the node is high or the value of the resulting cocluster is high, the node is removed from the cocluster.

Coclustering framework using Genetic Algorithm (GA): Coclustering approach is viewed as optimization problem with the objective of discovering overlapping coclusters with high ACV and high volume. The important feature of GA (Chakraborty and Maka, 2005) is that it provides a

number of potential solutions to a given problem and the choice of final solution is left to the user. In this study, coclustering algorithm using genetic algorithm is proposed that embeds greedy algorithm as local search procedure to find highly correlated coclusters.

Initialization: Usually, GA is initialized with the population of random solutions. In order to avoid random interference, coclusters obtained from greedy search procedure are used to initialize GA. This will result in faster convergence compared to random initialization. Maintaining diversity in the population is another advantage of initializing with these coclusters.

Selection: Selection is an operation to select two parent strings for generating new offspring.

Crossover: Crossover is an operation where two parent strings exchange parts of their corresponding chromosomes.

Mutation: Mutation is an operation that changes one digit at a time. A digit is selected according to pre-defined mutation probabilities and replaced with a different number.

Evaluation: Evaluate the fitness of the population.

Fitness function: The main objective is to discover high volume coclusters with high ACV. The following fitness function is used to access the quality of cocluster:

$$F(u,p)=\begin{cases}|u|*|p|, \text{if ACV (cocluster) is greater than} \\ \text{ACV threshold } \delta \\ 0, \text{otherwise}\end{cases} \quad (5)$$

where, |u| and |p| are number of users and pages in cocluster and δ is defined as according to the problem or nature of the dataset.

Algorithm 2
Coclustering framework using genetic algorithm

Input: Set enlarged and refined seed
Output: Optimal coclusters

```
Set t = 0, max_iteration and ACV threshold δ;
 Initialize the population P using seeds obtained from the Two-Way
K-Means Clustering or Greedy Search Procedure.
while (t<= max_iteration) do
// Select individuals for reproduction selection (P)
// Recombine individuals (crossover)
Crossover (P)
//Apply Mutation
Mutate (P)
//Calculate the fitness of offspring
Evaluate fitness of P
Reinsert offspring into population
t = t + 1
end
Return the optimized coclusters
```

EXPERIMENTAL RESULTS AND ANALYSIS

The experiments are conducted on the well-known benchmark clickstream dataset called msnbc dataset which was collected from MSNBC.com portal. This dataset is taken from UCI repository where the original data is preprocessed using Eq. 1. There are 989,818 users and only 17 distinct items because these items are recorded at the level of URL category, not at page level which greatly reduces the dimensionality.

The length of the clickstream record starts from 1-64. Average number of visits per user is 5.7. Intuitively, user sessions contain very small and very large number of visited URL category may not provide any useful information about the user's behavior. Therefore, during the log data preprocessing step, a filtering process is applied to remove those user sessions.

K-means clustering using GA with two different fitness functions was implemented on msnbc dataset and results were shown in Table 2. It shows that fitness function-I based on intra-cluster distance based correlation measure is not able to extract the highly correlated user groups which is evident from their average ACV is 0.3200.

Whereas the fitness function-II which is based on average correlation value is able to extract somewhat better correlated user groups than fitness function I but not their average ACV close to 1. This is because when a clustering method is used for grouping users, it typically partitions users according to their similar browsing interest under all pages of a web site.

As said earlier, web users behave similarly only on a subset of pages and their behavior is uncorrelated over the rest of the pages. Therefore, from this study it is inferred that clustering methods failed to extract the highly correlated users groups for the entire set of pages. In addition, fitness function II is better than fitness function I in extracting the correlated user groups.

Table 2: Performance of clustering using GA

Characteristics	Fitness function (F-I)	Fitness function (F-II)
Average ACV	0.3200	0.6012
Average volume	8400.4	5940.6
Overlapping degree	0.1010	0.0089

Table 3: Performance of coclustering using greedy search procedures

Characteristics	Two-way K-means clustering	Coclustering using greedy search procedure
No. of coclusters	120	120
Average ACV	0.5711	0.8941
Average volume	694.3	1699.8
Overlapping degree (R index)	0	0.0202

Table 4: Performance of coclustering using GA

Population size	Average volume	Average ACV	Row (%)	Column (%)	Overlapping degree
120	11715	0.9709	99.9	82.35	0.2152

Table 3 shows the number of coclusters, average volume, average ACV and overlapping degree between the coclusters for each method. It is obvious from the observed values of average ACV and average volume as shown in the Table 3, Greedy search procedure extracts the optimal coclusters in terms of their volume and ACV than the Two-Way K-Means clustering by exploring the search space. Moreover, it has ability to identify the overlapping coclusters whose R index is 0.0202. Nevertheless, most of the times, it is stuck at local optimal solution. To overcome this problem, Coclustering using GA is designed and implemented to obtain global optimal coclusters using the correlated based fitness function.

From the results in Table 4, it is obvious that it correlates the relevant users and pages of a web site with a high degree of homogeneity (i.e., ACV). The performance values of coclusters using GA indicate that GA performs well in extracting the global optimal correlated cocluster whose average ACV is 0.9707 (close to 1), average volume is 11715 (which is greatly larger than coclustering using greedy search procedure). In addition, user coverage and page coverage by these coclusters are also good and overlapping degree among these coclusters is 0.2152. From the observed values as shown in the Table 4, it is inferred that coclustering using GA is significant in extracting the highly overlapped optimal correlated coclusters with shifting and scaling browsing patterns on a subset of the dimensions (pages) from the preprocessed web usage data.

In clickstream analysis, the frequency of visiting the pages of a web site of two users may rise or fall synchronously in response to a set of their interest. Though the magnitude of their interest levels may not be close but the pattern they exhibit can be very much similar. The proposed coclustering framework is interested in finding overlapping coclusters containing both scaling and shifting patterns and with a general understanding of users' browsing interest.

Figure 2 shows the comparison of Two-Way K-Means clustering, coclustering using greedy search procedure and coclustering using GA on msnbc dataset.

Table 5: Comparison of clustering and coclustering using GA

Parameters	Clustering using GA: Fitness function-I	Clustering using GA: Fitness function-II	Coclustering using GA
Average ACV	0.3200	0.6012	0.9709
Average volume	8400.4	5940.6	11715

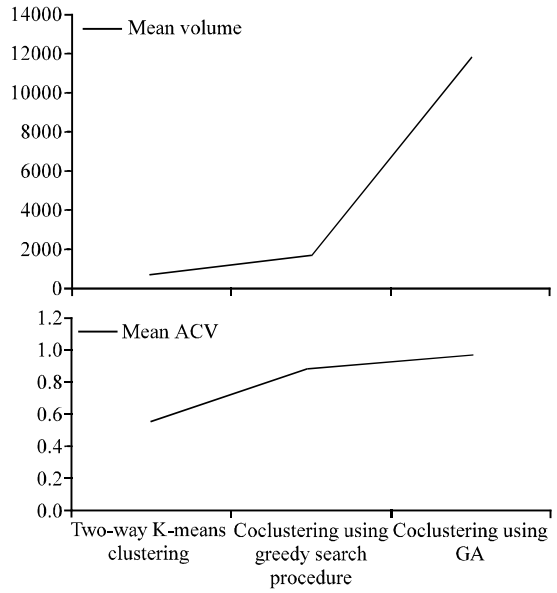

Fig. 2: Performance of coclustering algorithms

From this Fig. 2, we can know that coclustering using GA always outperforms other two coclustering methods for finding highly correlated coclusters with high volume (Table 5).

Coclustering framework using GA performs better than the clustering algorithms in extracting the highly correlated user groups from web usage data. With these coclustering results, one will be able to predict which kind of user is navigating although, we do not have any registering data about the web users. This information can be also used for marketing purpose, customizing the web site once we know the kind of user through the navigation characteristics.

CONCLUSION

Web clustering is an approach for grouping web users or pages into various categories according to underlying relationships among them. In this study, a new coclustering framework for clickstream data has been presented to extract the highly correlated coclusters of web objects using GA. GA is used as a global optimizer that can be coupled with a greedy search procedure which significantly increase the size of the largest homogeneity cocluster. The Average Correlation Value (ACV) has used

as a merit function with the aim of obtaining correlated coclusters with shifting and scaling browsing patterns. Results show that this approach achieves significant improvements in the quality of the coclusters when compared to the application of the greedy strategy alone. Future research will focus on some improvements for the proposed algorithm with regard to the fitness function.

REFERENCES

Bagyamani, J. and K. Thangavel, 2010. SIMBIC: Similarity based biclustering of expression data. Commun. Inform. Process. Manage., 70: 437-441.

Ben-Dor, A., B. Chor, R. Karp and Z. Yakhini, 2003. Discovering local structure in gene expression data: The order-preserving submatrix problem. J. Comput. Biol., 10: 373-384.

Busygin, S., G. Jacobsen and E. Kramer, 2002. Double conjugated clustering applied to leukemia microarray data. Proceedings of the 2nd SIAM International Conference on Data Mining/Workshop on Clustering High Dimensional Data (DM'02), Arlington, VA, USA., pp: 1-9.

Chakraborty, A. and H. Maka, 2005. Biclustering of gene expression data using genetic algorithm. Proceedings of the Computational Intelligence in Bioinformatics and Computational Biology, November 14-15, 2005, IEEE, pp: 1-8.

Cho, H., I.S. Dhillon, Y. Guan and S. Sra, 2004. Minimum sum-squared residue co-clustering of gene expression data. Proceedings of the 4th SIAM International Conference on Data Mining, April 2004, Lake Buena Vista, Florida, pp: 1-12.

Cooley, R., J. Srivastava and M. Deshpande, 1999. Data preparation for mining world wide web browsing patterns. Knowledge Infor. Syst., 1: 5-32.

Das, C., P. Maji and S. Chattopadhyay, 2008. A novel biclustering algorithm for discovering value-coherent overlapping σ biclusters. Proceedings of the 16th International Advance Computing Communications, December 14-17, 2008, Chennai, pp: 148-156.

Dhillon, I.S., 2001. Co-clustering documents and words using bipartite spectral graph partitioning. Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 29-29, 2001, ACM, New York, 269-274.

Dhillon, I.S., S. Mallela and D.S. Modha, 2003. Information-theoretic co-clustering. Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 24-27, 2003, Washington, DC, USA, pp: 89-98.

Getz, G., E. Levine and E. Domany, 2000. Coupled two-way clustering analysis of gene microarray data. PNAS, 97: 12079-12084.

Guandong, X., Y. Zong, P. Dolog and Y. Zhang, 2010. Co-clustering analysis of weblogs using bipartite spectral projection approach. Proccedings of the Knowledge-Based and Intelligent Information and Engineering Systems, November 8-9, 2010, UK., pp: 398-407.

Hartigan, J.A., 1972. Direct clustering of a data matrix. J. Am. Stat. Assoc. 67: 123-129.

Kluger, Y., R. Basri, J.T. Chang and M. Gerstein, 2003. Spectral biclustering of microarray data: Coclustering genes and conditions. Genome Res., 13: 703-716.

Koutsonikola, V.A. and A. Vakali, 2003. A fuzzy bi-clustering approach to correlate web users and pages. Int. J. Know. Web Intell., 1: 3-23.

Lee, C.H. and Y.H. Fu, 2008. Web usage mining based on clustering of browsing features. Proceedings of the 8th International Conference on Intelligent Systems Design and Applications, November 26-28, 2008, IEEE Computer Society, Washington DC., USA., pp: 281-286.

Mobasher, B., H. Dai, M. Nakagawa and T. Luo, 2002. Discovery and evaluation of aggregate usage profiles for web personalization. Data Min. Knowl. Discovery, 6: 61-82.

Rana, S., S. Jasola and R. Kumar, 2010. A hybrid sequential approach for data clustering using k-means and particle swarm optimization algorithm. Int. J. Eng. Sci. Technol., 2: 167-176.

Rathipriya, R., K. Thangavel and J. Bagyamani, 2011. Evolutionary biclustering of clickstream data. Int. J. Comput. Sci. Issues, 8: 32-38.

Srivastava, J., R. Cooley, M. Deshpande and P.N. Tan, 2000. Web usage mining: Discovery and applications of usage patterns from web data. SIGKDD Explorations, 1: 12-23.

Sujatha, N. and K. Iyakutty, 2010. Refinement of web usage data clustering from k-means with genetic algorithm. Eur. J. Sci. Res., 42: 478-490.

Tang, C. and A. Zhang, 2001. Interrelated two-way clustering: An unsupervised approach for gene expression data analysis. Proccedings of the 2nd IEEE International Symposium on Bioinformatics and Bioengineering, March 4-6, 2001, State University of New York, Buffalo, pp: 41-48.

Xie, B., S. Chen and F. Liu, 2007. Biclustering of Gene Expression data using PSO-GA hybrid. Proceedings of the 1st International Conference on Bioinformatics and Biomedical Engineering, July 6-8, 2007, IEEE, Wuhan, pp: 302-305.

Xu, G., Y. Zhang and X. Zhou, 2005. A web recommendation technique based on probabilistic latent semantic analysis. WISE2005. LNCS, 3806: 15-28.