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 co-clusters 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 co-cluster of clickstream data is a local pattern such that users in co-cluster exhibit correlated browsing pattern through a subset of pages of a web site. This study aims to provide a new correlated co-clustering framework using genetic algorithm to identify overlapping correlated co-cluster 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 co-cluster, 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 co-clusters. These co-clusters 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
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 pagevector 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 co-clustering 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 co-cluster 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).

Table 1: Coclustering algorithms for microarray data analysis

<table>
<thead>
<tr>
<th>Coclustering algorithms</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cho et al. (2004)</td>
<td>K-means based biclustering algorithm was proposed that identifies m row clusters and n column clusters simultaneously while monotonically decreasing the mean square residue defined by Cheng and Church.</td>
</tr>
<tr>
<td>Tang and Zhang (2004)</td>
<td>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. Double Conjugated Clustering (DCC) was proposed which aims to discover biclusters with coherent values defined using multiplicative model of bicluster.</td>
</tr>
<tr>
<td>Basygin et al. (2002)</td>
<td>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.</td>
</tr>
<tr>
<td>Getz et al. (2000)</td>
<td>Hybrid evolutionary optimization algorithm based on particle swarm and Genetic algorithms was presented in this study to solve the biclustering problem.</td>
</tr>
<tr>
<td>Xie et al. (2007)</td>
<td>Coclustering algorithm using (2005) genetic algorithm was proposed that embeds greedy algorithm as local search procedure to find the best biclusters.</td>
</tr>
</tbody>
</table>

Guandong et al. (2010) presented an algorithm using 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 co-clustering 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.

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 to sessions. A session is a sequence of pageviews by a single user during a single visit s = {p1, p2, p3, ..., ps}. 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 aij of A (U, P) represents frequency of the user Uj of U visit the page Pj of P during a given period of time:

\[ a_{ij} = \begin{cases} \text{Hits}(U_j, P_j) & \text{if } P_j \text{ is visited by } U_j, \\ 0, & \text{otherwise} \end{cases} \]  

(1)

Where:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hits (Uj, Pj)</td>
<td>The count/frequency of the user Uj visiting page Pj</td>
</tr>
<tr>
<td>Uj</td>
<td>Aces the page Pj during a given period of time</td>
</tr>
</tbody>
</table>

169
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 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:

\[
R(B) = \max \left( \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} |r_{\text{row}_{ij}}|}{m - n}, \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} |r_{\text{col}_{ij}}|}{m - m} \right)
\]

(2)

\( r_{\text{row}_{ij}} \) - The correlation between row \( i \) and row \( j \)
\( r_{\text{col}_{ij}} \) - 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 (Rathripiya 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

\[
F1 = \sum_{i=1}^{n} \sum_{j=1}^{k} |U_{ij} - \mu_{ij}|
\]

where:

- \( n \) = The number of users
- \( k \) = The number of clusters respectively
- \( U_{ij} \) = The user \( i \)
- \( \mu_{ij} \) = The cluster centre

**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:

\[
F2(u, p) = \begin{cases} |u|^p |p| & \text{if ACV(cluster)i is } > \text{ACV threshold } \delta \\ 0 & \text{Otherwise} \end{cases}
\]

where, \(|u|\) and \(|p|\) are number of users and pages in cluster and ACV threshold \( \delta \) 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 co-clusters. The degree of overlapping among all biclusters is defined as follows:

\[ R = \frac{1}{|U||P|} \sum_{i=1}^{P} \sum_{j=1}^{P} T_{ij} \]  

(4)

Where:

\[ T_{ij} = \frac{1}{(N-1)} \left( \sum_{i=1}^{n} a_{ij} - 1 \right) \]

Where:

- \( N \) = The total number of co-clusters
- \(|U|\) = The represents the total number of users
- \(|P|\) = The represents the total number of pages in the data matrix \( A \)

The value of \( w_i (a_{ij}) \) is either 0 or 1. If the element (point) \( a_{ij} \) is in the \( k \)th co-cluster then \( w_i (a_{ij}) = 1 \), otherwise 0. Hence, the \( R \) index represents the degree of overlapping among the co-clusters. If \( R \) index value is higher, then degree of overlapping of the generated co-clusters would be high. The range of \( R \) index is \( 0 \leq R \leq 1 \).

**Coclustering Framework for Clickstream Data Using Genetic Algorithm**

In this research, co-clusters with highly correlated users and high volume are preferred. Therefore, the fitness function based on ACV used to evaluate the quality of co-clusters first step is to identify the initial co-clusters 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 co-clusters 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 co-clusters from clickstream data patterns. These overlapped co-clusters 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 co-clusters.

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 \( c_u^{i} \) be a subset of users and \( c_p^{j} \), \( C_p^{j} (1 \leq j \leq k_p) \). Let \( c_u^{i} \) be a subset of pages and \( c_p^{j} \), \( C_p^{j} (1 \leq j \leq k_p) \). The pair \( (c_u^{i} , c_p^{j}) \) denotes a co-cluster of \( A \). By combining the results of user dimensional clustering and page dimensional clustering, \( k_u \times k_p \) co-clusters are obtained. These correlated co-clusters are called seeds. The proposed co-clustering 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 co-cluster 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 co-clusters 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 grows up incrementally. After seed enlargement, enlarged seeds are refined on the both dimensions to get optimal co-clusters.

**Algorithm 1**

**Coclustering using greedy search procedure:**

Input: User access matrix \( A \)

Output: Set of enlarged and refined co-clusters

1. Compute \( k_u \) user clusters and \( k_p \) page clusters from preprocessed clickstream data
2. Combine \( k_u \) and \( k_p \) clusters to form \( k_u \times k_p \) co-clusters called seeds
3. For each seed do
   1. Add/remove the element (user/page) to/from the co-cluster which increases the ACV of seed
4. End for
5. Return enlarged and refined co-clusters

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 co-cluster. Either incremental value of ACV after adding the node is high or the value of the resulting co-cluster is high, the node is removed from the co-cluster.

**Coclustering framework using Genetic Algorithm (GA):**

Coclustering approach is viewed as an optimization problem with the objective of discovering overlapping co-clusters 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, co-clustering 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:

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

where, \(|u|\) and \(|p|\) are number of users and pages in cocluster and \(\delta\) 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 \(t = 0\), \(max_{\text{iteration}}\) and \(ACV\) threshold \(\delta\).

Initialize the population \(P\) using seeds obtained from the Two-Way K-Means Clustering or Greedy Search Procedure.

while \((t < max_{\text{iteration}})\) 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>
<thead>
<tr>
<th>Table 2: Performance of clustering using GA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Characteristics</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>Average ACV</td>
</tr>
<tr>
<td>Average volume</td>
</tr>
<tr>
<td>Overlapping degree</td>
</tr>
</tbody>
</table>
Table 3: Performance of co-clustering using greedy search procedures

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Two-way K-means clustering</th>
<th>Co-clustering using greedy search procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of clusters</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>Average ACV</td>
<td>0.5711</td>
<td>0.8941</td>
</tr>
<tr>
<td>Average volume</td>
<td>694.3</td>
<td>1699.8</td>
</tr>
<tr>
<td>Overlapping degree (R index)</td>
<td>0</td>
<td>0.0202</td>
</tr>
</tbody>
</table>

Table 4: Performance of co-clustering using GA

<table>
<thead>
<tr>
<th>Population size</th>
<th>Average volume</th>
<th>Average ACV</th>
<th>Overlapping degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>11715</td>
<td>0.9709</td>
<td>99.9</td>
</tr>
</tbody>
</table>

Table 5: Comparison of clustering and co-clustering using GA

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Clustering using GA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fitness function-I</td>
<td>Fitness function-II</td>
</tr>
<tr>
<td>Average ACV</td>
<td>0.3200</td>
</tr>
<tr>
<td>Average volume</td>
<td>8400.4</td>
</tr>
</tbody>
</table>

Table 3 shows the number of co-clusters, average volume, average ACV and overlapping degree between the co-clusters 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 co-clusters 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 co-clusters whose R index is 0.0202. Nevertheless, most of the times, it is stuck at local optimal solution. To overcome this problem, Co-clustering using GA is designed and implemented to obtain global optimal co-clusters 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 co-clusters using GA indicate that GA performs well in extracting the global optimal correlated co-cluster whose average ACV is 0.9707 (close to 1), average volume is 11715 (which is greatly larger than co-clustering using greedy search procedure). In addition, user coverage and page coverage by these co-clusters are also good and overlapping degree among these co-clusters is 0.2152. From the observed values as shown in the Table 4, it is inferred that co-clustering using GA is significant in extracting the highly overlapped optimal correlated co-clusters 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 co-clustering framework is interested in finding overlapping co-clusters 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, co-clustering using greedy search procedure and co-clustering using GA on msnbc dataset.

Fig. 2: Performance of co-clustering algorithms

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

Co-clustering framework using GA performs better than the clustering algorithms in extracting the highly correlated user groups from web usage data. With these co-clustering 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 co-clustering framework for clickstream data has been presented to extract the highly correlated co-clusters 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 co-cluster. The Average Correlation Value (ACV) has used
as a merit function with the aim of obtaining correlated co-clusters with shifting and scaling browsing patterns. Results show that this approach achieves significant improvements in the quality of the co-clusters 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


