Computer and Information Science
www.ccsenet.org/journal.html
Vol. 1, No. 3
August 2008

Similarity Matrix Based Session Clustering by Sequence Alignment Using Dynamic Programming

Dr.K.Duraiswamy
Dean, Academic
K.S.Rangasamy College of Technology
Tiruchengode, India

V. Valli Mayil (Corresponding author)
P.G.Department of Computer Science
Kongu Arts and Science College, Erode, India
Tel: 91-0424-233-9149 E-mail: vallimayilv@gmail.com

Abstract

With the rapid increasing popularity of the WWW, Websites are playing a crucial role to convey knowledge to the end users. Every request of Web site or a transaction on the server is stored in a file called server log file. Providing Web administrator with meaningful information about user access behavior (also called click stream data) has become a necessity to improve the quality of Web information and service performance. As such, the hidden knowledge obtained from mining, web server traffic data and user access patterns (called Web Usage Mining), could be directly used for marketing and management of E-business, E-services, E-searching , E-education and so on.

Categorizing visitors or users based on their interaction with a web site is a key problem in web usage mining. The click stream generated by various users often follows distinct patterns, clustering of the access pattern will provide the knowledge, which may help in recommender system of finding learning pattern of user in E-learning system , finding group of visitors with similar interest , providing customized content in site manager, categorizing customers in E-shopping etc.

Given session information, this paper focuses a method to find session similarity by sequence alignment using dynamic programming, and proposes a model such as similarity matrix for representing session similarity measures. The work presented in this paper follows Agglomerative Hierarchical Clustering method to cluster the similarity matrix in order to group similar sessions and the clustering process is depicted in dendrogram diagram.

Keywords: Clustering, Sequence alignment, Dynamic Programming, Aggelomerative Clustering, Similarity matrix, Preprocessing, Session

1. Introduction

The World Wide Web is continuously growing with the large volume of transaction information and access request of web server. The task of obtaining hidden knowledge from web log file is called Web Usage Mining. As such, the enormous information in log file, it can not be used directly to obtain the knowledge. Data mining from web access log is a process consisting of three consecutive steps: (1) data gathering and pre-processing for filtering and formatting the log entries,(2) pattern discovery which consists of the use of a variety of algorithms such as association rule mining, sequential pattern analysis, clustering and classification on the transformed data in order to discover relevant and potentially useful patterns, and finally,(3) pattern analysis during which the user retrieves and interprets the patterns discovered.

The work in this paper follows sequence of processes as follows. Identification of session information from server log file is one of the important tasks in preprocessing of log file. Session is nothing but a sequence of web pages accessed in a period of time. Continuous click of web pages is identified as a session. Clustering of user session starts with finding similarity between all pairs of user sessions. Here we use the sequence alignment using dynamic programming technique to find session similarities. A matrix is constructed for maintaining session similarity values and it is clustered based on Agglomerative technique. The knowledge obtained after grouping similar web session's is used to group the visitors of the system who has same access behavior in a system.

1.1 Related work

Most of the studies in the area of web usage mining are very new, and the topic of clustering web sessions has recently become popular in the field of real application of clustering techniques. Shahabi et al. [5] introduced the idea of Path Feature Space to represent all the navigation paths. Similarity between each two paths in the Path Feature Space is measured by the definition of Path Angle which is actually based on the Cosine similarity between two vectors. In this work, k-means cluster method is utilized to cluster user navigation patterns. Fu et al. [4] cluster users based on clustering web sessions. Their work employed attribute oriented induction to transfer the web session data into a space of generalized sessions, then apply the clustering algorithm BIRCH [6] to this generalized session space. Most of the previous related works apply either Euclidean distance for vector or set similarity measures, Cosine or Jaccard Coefficient. Shortcomings for doing this are obvious. Another method of grouping the session is finding similarity measure between sessions and clusters them accordingly. Fig 1.1 discusses the sequence of procedures of grouping the session

2. Session Similarity Measure by sequence alignment

As the Web log contains data of every request of the server, it is to be preprocessed to obtain the relevant data, consists of sequence of sessions. The first and foremost question needed to be answered in clustering web sessions is how to measure the similarity between two web sessions. A web session is naturally a stream of hyper link clicks. Here we consider the original session data as a set of sequences, and apply sequence alignment method to measure similarity between sessions.

FastLSA [8] is a dynamic programming algorithm produces the optimal alignment for a given scoring function. We use sequence alignment techniques to analyze the sequence of user requests that appear in user sessions. For two strings of length *m* and *n*, optimal sequence alignment has zero or more gaps inserted into the sequence to maximize the number of positions in the aligned strings that match.

2.1 Session Similarity Measure by Dynamic Programming

Following data cleaning and preprocessing steps, the session similarity measure by dynamic programming is used in calculating the similarities between all pairs of session. Since user sessions are ordered URL requests, the session is referred as sequences of Web pages. The problem of finding the optimal sequence alignment is solved using dynamic programming approach [7,3,2]. Briefly, the algorithm consists of three steps. The first step is initialization where a dynamic programming matrix is created with K+1 columns and N+1 rows where K and N correspond to the sizes of the sequences to be aligned. One sequence is placed along the top of the matrix (sequence#1) and the other one along the left-hand-side of the matrix (sequence#2). A gap is added to the start of each sequence which indicates the starting point of calculation of similarity score.

To translate a matrix path to an alignment, follow the path from the top left to the bottom right. Every diagonal move corresponds to aligning two letters as either a match or a mismatch. A right move corresponds to the insertion of a gap in the vertical sequence. Down move corresponds to the insertion of a gap in the horizontal sequence. Any path can be translated to an alignment, Consider the session sequences as in table 2.1.

Considering first two sequences from table 2.1

S1=P1,P5,P7,P3,P6
S2= P1,P5,P3,P6

The alignment process, such as gap may be inserted between sequences S1 & S2 in order to make it match

S1 = P1	P5	P7	P3	P6
S2 = P1	P5	-	P3	P6

The alignment path matrix is given in the following table 2.2

Two sequences are scanning from left to right and the corresponding arrow is placed in the table.

Diagonal arrow - Match or mismatch of pair of web pages from sequences

Right arrow gap on top

Down arrow - gap on left

In order to obtain the optimal alignment for a given scoring function, we need to identify the corresponding *optimal path*. To derive the optimal path in the matrix, the algorithms can be divided into two phases, which we call *FindScore* and *FindPath*. Table 2.3 shows the DPM (Dynamic Programming Matrix) scores for the example sequences that are computed during the *FindScore* phase. The entries with numerical subscripts form the optimal path, which is computed in the *FindPath* phase.

In the *FindScore* phase, a 0 is placed in the upper-left corner of the matrix. Each algorithm propagates scores from the upper-left corner of the matrix to the lower-right corner of the matrix. The score that ends up in the lower-right corner is the optimal alignment score.

The score of any entry is the maximum of the three scores that can be propagated from the entry on its left, the entry above it and the entry above-left + Scoring function. For every pair of identical pages a positive score of value 20 is given as scoring function and for mismatch or gap negative value -10 is given as scoring function. For Example the score of 20_5 in (P1/P1) entry is the maximum of the score from its left entry (-10 + -10), above entry (-10 + -10), and above left entry (0 + 20) is assigned. A path was read from the top left to the bottom right. The entry in lower right corner is the value of optimal score. In this example the optimal score value is 70. Following algorithm discusses the methods to obtain DPM and optimal score.

```
Algorithm: Optimal similarity score value for 2 sessions
        Input:  Pair of session sequences S1,S2
        Output: Optimal Similarity score Value
1 Matrix DPM is created with K+1 columns and N+1 rows where K and N correspond to the sizes of S1 & S2 sequences respectively
2 Align the sequence by providing gap in between the sequence so two sequences can be matched as much as possible
3 Place sequence S1 in top of Matrix and sequence S2 on left side of Matrix
4 Assign top left corner of matrix = 0;
5 Find optimal path
        a) Find score
            Find DPM with every cell entry = max (left entry, right entry, above left Entry + scorevalue)
                Where score value = 20 for pair of matching pages
                Score value = -10 for mismatch pages.
        b)Find path
For every 2 sequences the pair of web pages is read and arrow is placed as per the following rule.
            Construct the arrow as follows
                Diagonal arrow - Match or mismatch
                Right arrow gap on top
                Down arrow - gap on left
6 Traversing from top left to lower right corner of DPM , the optimal score value is
  available at lower right corner
```

2.2 Session Similarity measure

Session similarity measure [3] has two components, such as *alignment score* component and *local similarity component*. The alignment score component computes how similar the two sessions are in the region of their overlap. If the highest value of the score matrix of two sessions, s1 & s2, is v, and the number of matching pages is M, matching score value is s (m), then the alignment score component Sa is:

$$Sa(s1,s2) = v/ (s(m) * M)$$

This value is normalized by the matching score and the number of matching pages. The local similarity component computes how important the overlap region is. If the length of the aligned sequences is L, the local similarity component is

$$Sb = M/L$$

Then the overall similarity between two sessions is given by

$$sim(S1,s2) = Sa * Sb.$$

For the above example the similarity measure of two session S1 & S2 is 0.7

$$Sim(S1,S2) = 70/(20*5)= 0.7$$

In this way, Similarity value of all pairs of session sequences of table 2.1 are found and depicted in table 2.4 & table 2.5. The highest value of two pair of session defines the more similarity between them and lowest value shows the dissimilarity between the pages.

3. Agglomerative hierarchical Clustering

Clustering is the process of grouping objects into clusters such hat the objects from the same cluster are similar and objects from different clusters are dissimilar. Similarity matrix obtained in previous section with largest value has high similarity whereas the less number have low similarity.

The objective of clustering in similarity matrix is to group the data with similar characteristics. In this work the sessions are arranged in similarity matrix and extracting knowledge from them, groups the similar session together. In this work Agglomerative hierarchical clustering process is used to cluster the similarity matrix. The process starts by finding the clusters with the high similarity value and putting those clusters into one cluster. The hierarchy of clusters formed can be represented by a dendrogram. Single linkage nearest neighbor technique is used across cluster to determine the similarity between the clusters.

Consider the array of similarity as given in table 2.4. The highest similarity value in the matrix is 70, available for S1 & S2, and at first stage sessions S1, S2 are combined to form second stage of matrix as follows.

The next highest similarity value is 50 for the sessions S4, S5. Now sessions S4, S5 are combined and clusters (S1,S2) & (S4,S5) are formed. The session S1, S2, S3 are combined, with the session similarity value 40, and (S1, S2, S3), (S4, S5) clusters are available; finally these two clusters are formed to produce resultant clusters. The dendrogram of clustering process is shown in fig 3.1

```
Algorithm : Agglomerative Hierarchical clustering
                    Input : Similarity matrix
                    Output : Dendrogram cluster
1  Select the largest similarity value from matrix and its session is Si,Sj and combine and form its
                composition Si,j
2  Form a matrix with Si,j
3  Find the cell value of matrix as
                Similarity(Si,j, Sk) = min { similarity (Si,Sk), Similarity(Sj,Sk)}
4  Repeat   step 2 until single cluster   in matrix cell.
```

4. Conclusion

Categorizing visitors or users based on their interaction with a web site is a key problem in web usage mining. This paper focuses on clustering the session details obtained from logfile. Session is a sequence of web pages accessed within a period. Session details are obtained after the preprocessing of log file. The work presented in this paper finds the similarity between every pair of session by sequence alignment using dynamic programming and similarity matrix is clustered using Agglomerative hierarchical technique in order to group the same similarity session. The objective of this process is to create a recommender system which will be used in finding group of visitors of session with the same nature in E-learning system, finding group of visitors with similar interest, categorizing customers in E-shopping etc or group of customers' session in marketing environment.

References

A. Banerjee and J. Ghosh. (2001). Clickstream clustering using weighted longest common subsequences. In Proc. of Workshop on Web Mining in First International SIAM Conference on Data Mining, pages 33–40, Chicago, April 2001.

Aaron Davidson A Fast Pruning Algorithm for Optimal Sequence Alignment.

S¸. G¨und¨uz and M. T. ¨ Ozsu. (2003). A web page prediction model based on click-stream tree representation of user behavior. In Proceedings of Ninth ACM International Conference on Knowledge Discovery and Data Mining (KDD), August 2003.

Y. Fu, K. Sandhu, and M.-Y. Shih. (1999). Clustering of web users based on access patterns. *WEBKDD* workshop, 1999.

C. Shahabi, A. Zarkesh, J. Adibi, and V. Shah. (1997). Knowledge discovery from users web-page navigation. *In workshop on Research Issues in Data Engineering,* England.

T. Zhang, R. Ramakrishnan, and M. Livny. (1996). BIRCH: an efficient data clustering method for very large databases. *In ACM SIGMOD*, pages 103–114, June 1996.

Weinan Wang Osmar R. Za¨ıane Clustering Web Sessions by Sequence Alignment.

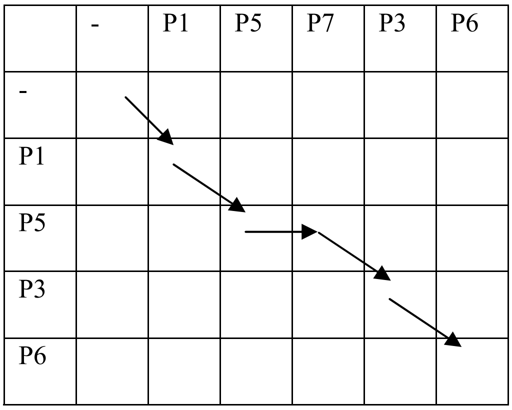
K. Charter, J. Schaeffer, and D. Szafron. (2000). Sequence alignment using fastlsa. In Proceedings of the 2000 International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences (METMBS’2000), pages 239–245.

Karypis, E.-H. Han, and V. Kumar. (1999). Chameleon: A hierarchical clustering algorithm using dynamic modeling. *IEEE Computer,* 32(8):68–75, August 1999.

Table 2.1 Session sequences

Session number	Sequences
S1	p1,p5,p7,p3,p6
S2	p1,p5,p3,p6
S3	p5,p7,p6
S4	p2,p8,p6,p5
S5	p2,p6,p5

Table 2.2 alignment path of session S1 & S2



	-	P1	P5	P7	P3	P6
-						
P1						
P5						
P3						
P6						

Table 2.3 DPM - optimal score value

	-	P1	P5	P7	P3	P6
-	0	-10	-20	-30	-40	-50
P1	-10	20_5	10	0	-10	-20
P5	-20	10	40_4	30_3	20	10
P3	-30	0	30	30	50_2	40
P6	-40	-10	20	20	40	70_1

Table 2.4 optimal score value of pair of sessions

	S1	S2	S3	S4	S5
S1	---	70	40	-50	10
S2	70	----	20	-20	-20
S3	40	20	----	-10	-10
S4	-50	-20	-10	---	50
S5	10	-20	-10	50	----

Table 2.5 Similarity measure Matrix

	S1	S2	S3	S4	S5
S1	1	0.7	0.4	-0.5	0.1
S2		1	0.25	-0.25	-0.25
S3			1	-0.25	-0.125
S4				1	0.625
S5					1

Table 2.6 Similarity matrices on applying single linkage rule

	S1,S2	S3	S4	S5
S1,S2	1			
S3	40	1		
S4	-20	-10	1	
S5	-20	-10	50	1

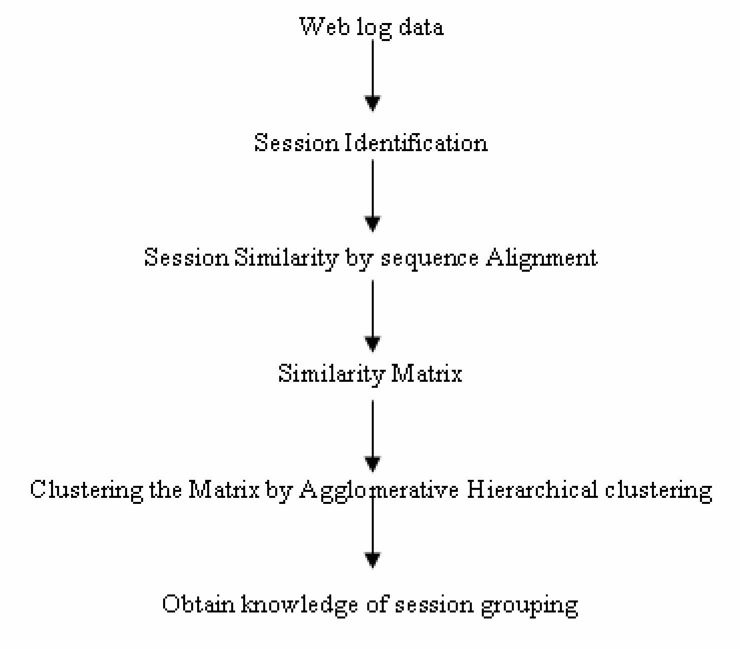

Web log data
Session Identification
Session Similarity by sequence Alignment
Similarity Matrix
Clustering the Matrix by Agglomerative Hierarchical clustering
Obtain knowledge of session grouping

Figure 1.1 procedures for grouping the session

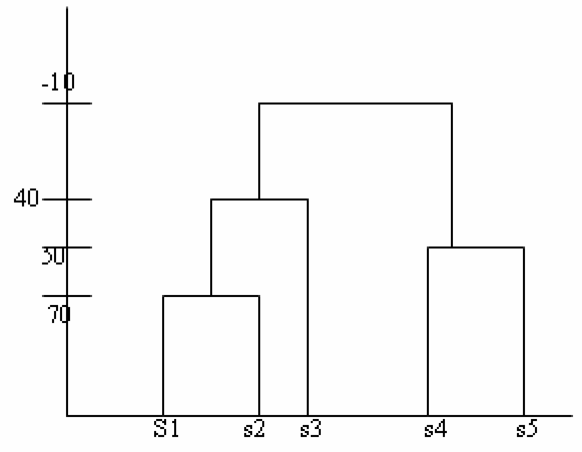


Figure 3.1 dendrogram for session clustering