A Graph-Based Approach for Discovering and Mining Evolving usage Patterns

M. Sree Vani
Dept of CSE, MGIT , Hyderabad-500075

Abstract

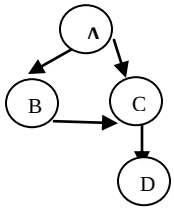
By monitoring web user's browsing behavior, we can discover reliable knowledge about user's general preferences and needs i.e. web usage patterns. Web usage patterns can help us to understand the different modes of usage and to know what kind of information the visitors seek and read on the web site and hoe this information evolves with time. Web usage pattern can be discovered based on browsing features using web usage mining techniques. Mining web usage patterns from web log files helps in making strong recommendations for how to retain and increase the visitors. In this paper, we present an efficient Graph based approach for discovering and mining evolving usage patterns from web log files. We perform clustering of the user sessions extracted from web logs to partition the users into several homogeneous groups with similar activities and then extract usage patterns from each cluster. We construct usage pattern graph to discover the common interest in each group of users. Usage pattern graphs are constructed for subsequent new periods of web logging to discover changes in the user's interest.

Key words : *Web usage mining, Usage Patterns, Clustering, Evolving usage patterns, usage pattern graph.*

I. INRODUCTION

The goal of web usage mining is to find out the useful information from web data or web log files. The other goals are to enhance the usability of the web information and to apply the technology on the web applications. The result of web usage mining can be used for target advertisement , improving web site design, improving satisfaction of customer, guiding the strategy decision of the enterprise, and marketing analysis etc[2][4][7][9]. Maintaining a website is just as important as building it. To maintain website we need to improve its design. To improve the design of website we should find out how it is used by analyzing user's browsing behavior[7].Analyzing the behavior of a website user's is a research field which consists in adapting the data mining methods to the records of access log files. These files collect data such as the IP address of the connected host, the requested URL, the date and other information regarding the navigation of the user. Web usage mining applies data mining methods to discover web usage pattern through web usage data. Item-set mining (Fig 1.a), Sequential pattern mining (fig 1.b), and Graph mining (Fig 1.c), are examples of data mining methods that can be used to analyze web usage data [18].

Fig 1 (a) Item-set **(b) subsequence**



(c) substructure

Usage patterns can be discovered using web usage mining techniques that can automatically extract frequent access patterns from web log files. These patterns can later be harnessed toward personalizing the website to the user or to support targeted marketing. Hence it was crucial to understand the different nodes of usage and to know what kind of information the visitors seek and read on the website and how this information avoids with time. In this paper, We perform clustering of the user sessions extracted from web logs to partition the users into several homogeneous groups with similar activities and then extract usage patterns from each cluster. We construct usage pattern graph to discover the common interest in each group of users. Usage pattern graphs are constructed for subsequent new periods of web logging to discover changes in the user's interest.

The rest of this paper is organized as follows. In section 2, we describe our approach to usage pattern discovery using web usage mining. In section 3, we discuss our approach for tracking evolving usage patterns. In section 4, we present our results in mining evolving user profiles. Finally, in section 5, we present our conclusion and future work.

II. USAGE PATTERNS DISCOVERY BASED ON BROWSING FEATURES

The framework for our Web usage mining and a road map to the rest of this paper is summarized in Figure. 1, which starts with the preprocessing of Web server logs includes data cleaning and sessionization, and then continues with the data mining/pattern discovery via clustering. This is followed by a post processing of the clustering results to obtain Web user profiles and finally ends with user profile evolution.

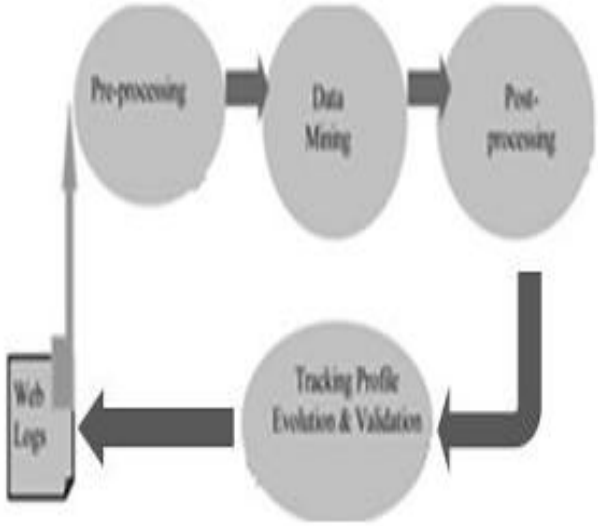


Figure:2 Web Usage Mining and user Profile Evaluation

The automatic identification of user profiles is a knowledge discovery task consisting of periodically mining new contents of the user access log files and is summarized in the following steps:

A. Extract User Sessions from Web log files
B. Similarity Measure used in clustering
C. Cluster the user sessions by using hierarchical agglomerative clustering method,
D. Post processing of session clusters into usage patterns

A. *Extract User Sessions from Web Log Files*

The first step in preprocessing consists of creating user sessions from web log files. User sessions can be reorganized as a m x k matrix as Table 1, each row can be presented by sessionu(Pu,1, Pu,2 ,..., Pu,k) where Pu,j =1 denotes that user u visited the web page otherwise Pu,j = 0. The k is the number of web pages. The session means the user's browsing situation which is also user's browsing feature.

Table 1. User Sessions

	P_1	P_2	…	P_k
$session^1$	1	0	…	1
$session^2$	0	1	…	0
⋮	⋮	⋮	…	⋮
$session^m$	1	1	…	1

B. *Similarity Measure Used in Clustering*

The similarity between any two users can be calculated by distance measure. Euclidean distance function (1) is used for computing the similarity between useri and userj, the similarity can be present by Sim (useri, userj) =(sessioni , sessionj) Euclidean distance is further normalized by equation (2). Further, the m x m matrix of user similarity will be obtained.

Euclidean distance :

$$D(user_i, uer_j) = \sqrt{\sum_{l=1}^{k} (p_{i,l} - p_{j,l})^2} \quad (1)$$

Normalization :

$$ND(user_i, uer_j) = 1 - \sqrt{\frac{\sum_{l=1}^{k} (p_{i,l} - p_{j,l})^2}{k}} \quad (2)$$

C. *Cluster the user Sessions by using Hierarchical Agglomerative Clustering Method*

In the hierarchical agglomerative clustering method, the distances are considered between centroids of clusters. The two clusters are merged by the shortest distance between two centroids. In the final, the new centroid vector of new cluster will be calculated by equation (3). In this paper, the single-linkage and complete-linkage are not considered, but distances of centroids are used. It is assumed there are n objects in a cluster, the feature of each object can be represented by (Pi,1 , Pi,2……Pi,k) where 1≤i≤n . The centroid vector of cluster can be calculated as follows:

$$centroid^{cluster} = \left(\frac{\sum_{l=1}^{n} p_{i,1}}{n}, \frac{\sum_{l=1}^{n} p_{i,2}}{n}, \ldots, \frac{\sum_{l=1}^{n} p_{i,k}}{n} \right) \quad (3)$$

Hierarchical agglomerative clustering applied into our model procedures are as follows:

(1) Initialization cluster:
(1.1) Each object be a cluster.
(1.2) Creating similarity matrix of users.

(2) Clustering:

(2.1) Finding a pair of the most similar clusters and merging.

(2.2) Computing the new centroid vector of new cluster.

(2.3) Computing the distances between new cluster and others.

(2.4) Pruning and updating the similarity matrix.

(2.5) If the terminal condition is satisfied then output, else repeating 2.1 to 2.4.

(3) Cluster output:

(3.1) Output index table.

(3.2)Output all clusters.

Table 2: User Clusters

	P_1	P_2	...	P_k
$cluster^1$	10	20	...	10
$cluster^2$	0	14	...	45
. . .	. . .	. . .	...	. . .
$cluster^n$	23	34	...	12

User clusters can be reorganized as a m x k matrix as Table 2, each row can be presented by clusteru (Pu,1, Pu,2 ,..., Pu,k) where Pu,j value denotes that in cluster u , among all sessions how many number of users are visited the web page. The k is the number of web pages. The cluster means the common usage patterns of user's browsing behavior.

D. Post Processing of Session Clusters into Usage Patterns

After automatically grouping sessions into different clusters, we summarize the session categories in terms of vectors [3], [4] pi. The kth component/weight of this vector (pik) captures the relevance of page k in the ith profile, as estimated by the conditional probability that page k is accessed in a session belonging to the ith cluster (this is the frequency with which page k was accessed sessions belonging to the ith cluster). The profiles are then converted to binary vectors (sets) so that only pages with weights > 0.15 remain. Each profile, pi is discovered along with an automatically determined measure of scale αi that represents the amount of variance or dispersion of the user sessions in a given cluster around the cluster representative (profile). This measure will later serve an important role in determining the boundary of each cluster and thus allows us to automatically determine whether two profiles are compatible or not.

Table 3: Binary User clusters

	P_1	P_2	...	P_k
$cluster^1$	1	0	...	1
$cluster^2$	0	1	...	0
. . .	. . .	. . .	...	. . .
$cluster^n$	1	1	...	0

III. USAGE PATTERN GRAPH

Each user pattern becomes the node of UPD (Usage Pattern Graph), and similarity between each pattern is edge between two nodes of the graph as shown in figure 2.

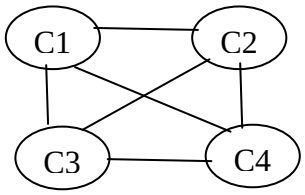


Figure 3: Usage Pattern Graph

The similarity between any two clusters can be calculated by distance measure. Euclidean distance function (4) is used for computing the similarity between clusteri and clusterj, the similarity can be present by Sim (profilei, profilej) =(clusteri , clusterj) Euclidean distance is further normalized by equation (5). Further, the m x m matrix of profile similarity will be obtained. And it serves as edge matrix.

$$D(profilei,profilej)=\sqrt{\sum_{l=1}^{k}(p_{i,l}-p_{j,l})^2} \quad (4)$$

$$ND(\ profilei,\ profilej\ )=1-\sqrt{\frac{\sum_{l=1}^{k}(p_{i,l}-p_{j,l})^2}{k}} \quad (5)$$

IV. MINING EVOLVING USAGE PATTERNS

Mining different pattern events across different time period can generate a better understanding of the evolution of user access patterns and seasonality. Each profile pi is discovered along with an automatically determined measure of scale αi that represents the amount of variance or dispersion of the user sessions in a given cluster around the cluster Representative. This measure is used to determine the boundary around each cluster and thus allow us to automatically determine whether two profiles are compatible. After mining the web log of a given period , we perform an automated comparison between all the profiles discovered in the previous batch by a sequence of SQL queries on the profiles that have been stored in a database , as shown in the "track profile" Algorithm.

Algorithm: TrackProfiles

Input:-Discovered Profiles for all Time Periods stored in Database
//(profile= set of relevant pages, and scale αi)

-Begining time period T1, Ending time period Tk.

Output:-Profile Trail: Profile-to-Profile tracking Table from T1 to Tk (e,g table 4)

for I=T1 to Tk do

for J=first profile in time period I to lasr profile in Time period I do

for K= first profile in Time period I+1 to last profile in Time period I+1 do
{
Distance[k] = Sweb(profilei,profile k);

if Distance[k] < αj then Insert into ProfileTrail(period,Thisprofile,TothisProfile)
values(I,Profile[j],k);
}

V. EXPERIMENTAL RESULTS

Then user browsed web page will be directed to Naïve Bayes classifier. Then we classified each browsed page summary and each ad with respect to the taxonomy. Following [11], we constructed additional features based on these immediate classifications as well as their ancestors in the taxonomy (the weight of each ancestor feature was decreased with a damping factor of 0.5). Each page and ad was represented as a bag of words (BOW) and an additional vector of classification features. Finally, the ad retrieval function was formulated as a linear combination of similarity scores based on both BOW and classification features:

$$score(page, ad) = \alpha \cdot simBOW(p, a) + \beta \cdot sim_{class}(p, a)$$

Where simBOW(p, a) and sim_{class}(p, a) are cosine similarity scores between page p and ad a using BOW and classification features, respectively. Then Ad recommendation system uses the above retrieve function to retrieve relevant ads from database scrolls the retrieved ads dynamically on user browsed web page. Thus, a user can be assigned a particular ad on a user interested webpage that generates a higher user response. By targeting particular campaigns to certain users on a given website, more advertising dollars will be generated. A snapshot of training sample is shown in Table 2.

Table 4: User Profiles Database

Term-No	Webpage#	$Term_1$	$Term_2$	…	$Term_n$	Cluster name
$Term_1$	http://sports.yahoo.com/	23	11	……	12	Sports
$Term_2$	http://www.health.com/	22	55	……	15	Health
….	……	….	…..	……	….	
$Term_n$	http://www.google.com/finance	12	11	……	45	Finance

VI. CONCLUSION

The user profile improves the search engine's performance by identifying the information needs for individual users. The user's positive preferences were inferred using the mining rules and utilized the preferences in deriving user's profiles. The user profiling strategies were evaluated and compared with the personalized query clustering method. The agglomerative clustering algorithm is employed for finding queries that are conceptual ly close to one another. The user profiles capturing both the user's positive and negative preferences perform the best among the user profiling strategies. The RSCF makes a search of data containing the item in the search results, th e required data is been clicked by the user and this clicked data is given as the input and generates the rankers as the output.

REFERENCES

[1] L.Becchetti et al. Link-based characterization and detection of web spam. In Proc. of AIRWeb '06, 2006.
[2] A.A. Benczur et al. Spamrank - fully automatic link spam detection. In Proc. of AIRWeb '05, 2005.
[3] C.Castillo et al. A reference collection for web spam. SIGIR Forum, 40(2), 2006.
[4] C.Castillo et al. Know your neighbors: Web spam detection using the web topology. In Proc. of SIGIR '07, 2007.
[5] J.Caverlee and L. Liu. Countering web spam with credibility-based link analysis. In Proc. of PODC '07, 2007.
[6] D.Fetterly, M. Manasse, and M. Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In Proc. of WebDB '04, 2004.
[7] A.Ntoulas et al. Detecting spam web pages through content analysis. In Proc. of WWW '06, 2006.
[8] S.Webb, J. Caverlee, and C. Pu. Characterizing web spam using content and http session analysis. In Proc. of CEAS '07, 2007.
[9] D.Cai ,S.Yu Block-based web search ,In proceedings of SIGIR'04
[10] S.Webb,James C , Calton Pu ,Predicting Web Spam with HTTP Session Information, In proceedings of CIKM'08.