Using Web Clustering for Web Communities Mining and Analysis

Yanchun Zhang, Guandong Xu
Centre for Applied Informatics
Victoria University, Vic 8001, Australia
{Yanchu.Zhang, Guandong.Xu}@vu.edu.au

Abstract

Due to the inherent correlation among Web objects and the lack of a uniform schema of web documents, Web community mining and analysis has become an important area for Web data management and analysis. The research of Web communities spans a number of research domains such as Web mining, Web search, clustering and text retrieval. In this talk we will present some recent studies on this topic, which cover finding relevant Web pages based on linkage information, discovering user access patterns through analyzing Web log files, co-clustering Web objects and investigating social networks from Web data. The algorithmic issues and related experimental studies will be addressed. Some research directions are also to be discussed.

1. Introduction

The popularity and development of web technology has made the Internet an important and popular application platform for disseminating and searching information as well as conducting business. However, due to the lack of uniform schema for web documents and the huge amount of web information available on the Internet, web users always find it is difficult to obtain the needed information from the Internet accurately and easily. Such demands have posed a lot of challenges to web researchers and engineers.

For the data on the web, it has its own distinctive features from the data in conventional database management systems. Web data usually exhibits the following characteristics [1]: the data on the web is huge in amount, distributed, heterogeneous, unstructured, and dynamic.

These features indicate that web data is a specific type of data different from the data resided in traditional database systems. As a result, it is necessary to introduce a new data management model and schema to address web data management and processing.

Web community has emerged as a new efficient web data structure to model web documents. Unlike the conventional database management in which data models and schemas are defined, web community, which is a set of web-based objects (documents and users) having own logical structures, is another effective and efficient approach to reorganize web-based objects, support information retrieval and implement various applications. Therefore, compared to the traditional database based data management strategy, web community centered data management systems provide more flexible and effective capabilities to handle the tasks like web search and knowledge discovery in web-based data management.

In the context of web community research, web community mining and analysis is one of the important and interesting topics, which draws a large amount of attention from not only academia but also industrial partners. The aim of web community mining and analysis is to discover the inter- and intra-relationships among various web objects, e.g. web pages and web users. For example, one interesting and useful topic on web community mining is to identify the relevance between various web pages via linkage analysis. As the hyperlink information between two web pages does convey the intrinsic closeness from the viewing point of content association. Then the discovered pages which are relevant in content would aggregate to represent various web page communities, which can benefit other web tasks or applications, such as web search or web directory.

To address web community mining and analysis, there are a variety of analytical approaches and techniques available from the research progress in the areas of database, data mining, machine learning as well as information retrieval. Many research studies have demonstrated the success of web research using web clustering techniques. Web clustering is one kind of data processing algorithms that aggregate web objects into various clusters based on the mutual similarity of web objects. Known from the research of data mining and machine learning, clustering is an

978-0-7695-3496-1/08 $25.00 © 2008 IEEE
DOI 10.1109/WIIAT.2008.415

efficient unsupervising or weak-supervising algorithm to deal with the analysis of the large-scale data sets objectively. The main advantage of data analysis by using clustering in comparison with other approaches such as classification is that no or less human knowledge or intervention needs except calculating the mutual distance between the processed subjects during the whole process of data analysis. Thus it can avoid the bias brought by the data analyst. In this paper, we focus on investigating web clustering for web community mining and analysis.

In the context of clustering, how to define an appropriate similarity to measure subjects is one difficulty that greatly affects the performance of partitioning various subjects and the quality of the generated clusters. Secondly, choosing an efficient partition strategy, i.e. clustering algorithm, is another big concern that researchers need to address. To date a large amount of research efforts have been contributed to the development of clustering. In particular, in the cases of web community mining and analysis, we need to handle the problems encountered in clustering web data accordingly. To model the data on the web, there are a number of data expressions, such as matrix, data sequence or bipartite graph etc. Due to the expression simplicity and operation feasibility, the matrix expression is often considered as a universal web data model widely used in the areas of web data managements.

To perform web clustering on web data which is in the form of matrix expression, similarity calculations are intuitively carried out on the rows or columns of the matrix to compute the mutual distance between each web object pair. Therefore this computation leads to one main concern of the high computational cost resulting from the high dimensional input space. For example, in the context of web usage mining, there are usually tens to hundreds of thousands sessions in web log files, which represent a very highly dimensional usage data space. Consequently, the high computational difficulty will be incurred when we directly utilize user sessions rather than pages as dimensions, on which we employ clustering technique, and the conventional clustering techniques such as distance-based similarity methods are not capable of tackling this type computation of high-dimensional matrix.

On the other hand, the similarity between objects of one type (e.g. user session) depends on a subset of the objects of the other types (e.g. web pages) rather than all the subjects of the other type. In other words, the clustering on various object types can not be performed independently of each other. The clusters of one type of objects are usually aggregated in association with another type of objects, i.e. co-clusters. For example, in information retrieval, co-clusters may represent some specific subtopics in a collection of documents, which consist of a number of documents containing a set of dominant key words; while in the cases of web access pattern analysis, co-clusters are considered as segments of web visitors, who exhibit similar interests on various web pages together with a number of significant web pages.

In this paper, we aimed to investigate employing web clustering for web community mining and analysis in terms of web page communities, web user access patterns and co-clusters of web pages and uses as well. The main contributions of this paper are described as follows:

(1). We proposed a web clustering approach for web linkage analysis via introducing a new correlation-based web page similarity. Experimental investigations have shown the findings of relevant page segments.

(2). We explored to identify the latent usage patterns by employing latent semantic analysis and clustering on web log files. User access patterns are shown by various user profiles.

(3). We aimed to address co-clustering in the context of web usage mining via using a tensor decomposition approach.

(4). We discussed the adapting of social network analysis in web community mining and analysis.

The rest of the paper is organized as follows: in Section 2, we describe using web clustering for web linkage analysis via a novel web page similarity, which helps to find web page communities with similar topological structures. Section 3 discusses how to combine latent semantic analysis and web clustering on web usage mining. Then in Section 4, we extend the current latent semantic analysis approaches to find the co-clusters of web objects from web log files via a three-way tensor decomposition, meanwhile, the issues of social network analysis on the web is also discussed in this section. We conclude this paper and outline the future research directions in section 5.

2. Finding Web Page Communities via Web Clustering

Web page ranking is an interesting and challenging topic in web community research with a great application potential for improving web information search. In the context of web information search, there are two famous algorithms commonly mentioned in literature, namely HITS and PageRank.

Hyperlink Induced Topic Search (HITS) is a representative of algorithms that reveals web page relationships conveyed by hyperlinks [2]. HITS algorithm is essentially a link-based approach that intends to find authority and hub pages from a link induced web graph. Authorities are those pages that provide the best source of information on a given topic, while hubs are those pages that provide collections of links to authorities. Because the computation of authority and hub pages is carried out at query time, it is not feasible for today's search engines, which need to handle tens of millions of queries per day.

The PageRank algorithm was originally proposed by Brin and Page and incorporated in the search engine Google [3]. In contrast to HITS, PageRank computes a single measure of quality for a page at crawl time. This measure is then combined with a traditional information retrieval (IR) score at query time, and has the advantage of much greater efficiency [4-7].

Today, Google has become a very popular search engine over the Internet due to its powerful search ability and acceptable retrieval accuracy. After the generation of the Google, many research efforts have been addressed to aim the improvement of recall rate and retrieval accuracy [8, 9]. In this section, we present one algorithm on web community analysis to address this topic.

2.1. Correlation-based Web Page Similarity

Web page clustering is an interesting topic in web community analysis, which aims to discover web page group sharing similar functionality or semantics. In this paper, we proposed a new web page clustering algorithm based on measuring page similarity in terms of correlation [10].

A web page similarity usually refers to a certain page space. Since we are concerned about clustering web-searched results in this work, we focus on a page space that is related to the user's query topics. This constructed page source is shown in Figure 1, where root page set *R* consists of *r* highest-ranked pages from the searched results, *BV* and *FV* are the back and forward vicinity sets of *R* respectively, *V* is the vicinity set of *BV* and *FV*, and *S* stands for the uniting set of *R*, *BV*, *FV* and original links between pages in *S*.

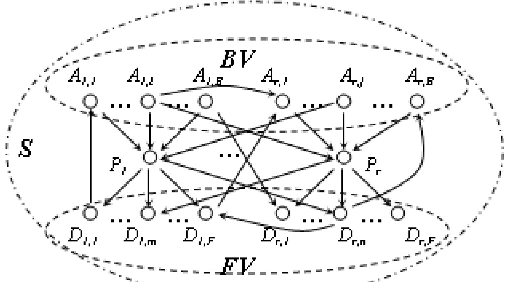


Figure 1. Structure of the page source S

For each web page, its correlation with other pages, via linkages, is expressed in two ways: one is out-links from it, another is in-links to it. In this algorithm, the similarity between two pages is measured by their own correlations with other pages in the page source *S*, rather than being derived directly from the links between them. More discussions and explanations with respect to definition of correlation are referred to [10]. Then, a new page similarity is introduced by the definition of page correlation degree with the concerned page source. For simplicity and better understanding of this new similarity, we divided the correlation matrix *C* into four blocks (sub-matrices) as follows.

$$C = (c_{ij})_{(m+n)\times(m+n)} = \begin{array}{c c} & \begin{array}{cc} R & V \end{array} \\ \begin{array}{c} R \\ V \end{array} & \left[\begin{array}{c:c} \textcircled{1} & \textcircled{2} \\ \hdashline \textcircled{3} & \textcircled{4} \end{array}\right] \end{array}$$

The elements in sub-matrix 1 represent the correlation relationships between the pages in *R*. Similarly, the elements in sub-matrices 2 and 3 represent the correlation relationships between the pages in *R* and *V*, and sub-matrix 4 gives the correlation relationships between the pages in *V*. It can be seen that the correlation degrees related with the pages in *R* are located in three sub-matrices 1, 2 and 3. Therefore, the similarity measurement for the pages in *R* only refers to the elements in these three sub-matrices.

In the correlation matrix *C*, the row vector that corresponds to each page *i* in *R* is in the form of

$$row_i = (c_{i,1}, c_{i,2}, \ldots, c_{i,m+n}),\ i = 1,2,\ldots,m \tag{1}$$

From the construction of matrix *C*, it is known that row_i represents *out-link* relationship of page *i* in *R* with all the pages in *S*, and element values in this row vector indicate the correlation degrees of this page to the linked pages. Similarly, the column vector that is in the form of

$$col_i = (c_{1,i}, c_{2,i}, \ldots, c_{m+n,i}), \quad i = 1,2,\ldots,m \tag{2}$$

represents *in-link* relationship of page *i* in *R* with all the pages in *S*, and its element values indicate the correlation degrees from the pages in *S* to page *i*.

Each page *i* in *R*, therefore, is represented as two correlation vectors: row_i and col_i. For any two pages *i* and *j* in *R*, their out-link similarity is defined as

$$sim_{i,j}^{out} = \frac{\left(row_i, row_j\right)}{\left\|row_i\right\| \cdot \left\|row_j\right\|} \tag{3}$$

where

$$(row_i, row_j) = \sum_{k=1}^{m+n} c_{i,k} c_{j,k} \ , \ \| row_i \| = (\sum_{k=1}^{m+n} c_{i,k}^2)^{1/2} \ .$$

Similarly, their *in-link similarity* is defined as

$$sim_{i,j}^{in} = \frac{\left(rol_i, rol_j \right)}{\left\| rol_i \right\| \cdot \left\| rol_j \right\|} \tag{4}$$

Then the similarity between any two pages *i* and *j* in *R* is defined as

$$sim\left(i, j\right) = \alpha_{ij} \cdot sim_{i,j}^{out} + \beta_{i,j} sim_{i,j}^{in} \tag{5}$$

where α_{ij} and β_{ij} are the weights for out-link and in-link similarities respectively.

With the page similarity measurement and the correlation matrix *C*, a web page clustering algorithm was employed to partition web pages into a number of page clusters. The details of the clustering algorithm are described as follows:

2.2. Algorithm Description

[**Algorithm**]: Similarity-*Based Web Page Clustering* (*R*, *T*)

[**Input**]: A set of web pages $R = \{p_1, p_2, \ldots p_m\}$, a clustering threshold *T*.

[**Output**]: A set of clusters $CL = \{CL_i\}$.

Step 1. Select the first page p_1 as the initial cluster CL_1 and the centroid of this cluster, i.e. $CL_1 = \{p_1\}$ and $CE = p_1$.

Step 2: For each page $p_i \in R$, calculate the similarity between p_i and the centroid of each existing cluster $sim\left(p_i, \ CE_j\right)$.

Step 3: If $sim\left(p_i, CE_k\right) = \max_j (sim(p_i, CE_j)) > T$, then add p_i to the cluster CL_k and recalculate the centroid CE_k of this cluster that consists of two vectors

$$CE_k^{row} = \frac{1}{| CL_k |} \sum_{j \in CL_k} row_j \ , \quad CE_k^{col} = \frac{1}{| CL_k |} \sum_{j \in CL_k} col_j \ , \tag{6}$$

where $|CL_k|$ is the number of pages in CL_k.

Otherwise, p_i itself initiates a new cluster and is the centroid of this new cluster.

Step 4: If there are still pages to be clustered (i.e. pages that have not been clustered or a page that itself is a cluster), go back to step 2 until all cluster centroids no longer change.

Step 5: Return clusters $CL = \{CL_i\}$.

2.3. Evaluation Results

Primary clustering experiments were conducted on a real web page source to evaluate the proposed page similarity measure. The page source was for the search topic "*Jaguar*". The search engine we used was *Google*. The number of pages in the root page set was 472, the total number of pages in the page source was 3,540, and the number of hyperlinks in the page source was 17,793. We present examples of some major clusters produced by the above algorithm in table 1. It is shown that the clustering results are satisfactory as all pages in the same cluster share the same topic. More simulation comparisons with other existing web clustering algorithms are discussed in [10].

Table 1 Examples of some major clusters

Topic: Jaguar Game
atarijaguardirectory.com // Atari Jaguar Directory
www.atarihq.com/interactive // Jaguar Interactive II
www.atari.org // The Definitive Atari Resource
Topic: Jaguar Big Cat
dspace.dial.pipex.com/agarman/jaguar.htm //Jaguar
www.animalsoftherainforest.com/jaguar.htm //Jaguar
www.bluelion.org/jaguar.htm // Jaguar
Topic: Jaguar Reef Touring
www.jaguarreef.com // Jaguar Reef Lodge
www.divejaguarreef.com // Dive Jaguar Reef Lodge
www.belizenet.com/jagreef.html // Jaguar Reef

3. Web Usage Mining via Web Clustering

In addition to finding relevant web pages with respect to the query web page by using linkage information conveyed by web pages, understanding web user navigational characteristics is another important aspect of web community analysis. To address this, web usage mining, one kind of the commonly used web data mining approaches, is proposed to discover web user behavior pattern and association between web pages from the viewing point of web user.

In the context of web usage mining, web clustering is one of the mostly used techniques to capture the aggregate property of web objects, such as web users or web pages. Generally, there are two kinds of clustering methods used in web usage mining, which are associated with the mined objects: user session clustering and web page clustering. Many web clustering studies have been conducted extensively and a variety of algorithms have been developed accordingly [5, 11]. One successful application of web page clustering is adaptive web site, for example, an algorithm called PageGather [12] is proposed to synthesize index pages that do not exist initially, based on finding web page segment sharing common semantic similarities.

To overcome the difficulty of the high computation, many solutions have been proposed to reduce the original high dimensional space by converting it into a transformed low-dimensional space, but keeping the minimum loss of the semantic information hidden in the web data. Latent Semantic Indexing (LSI) [13, 14] is one of the algorithms we can choose, which is to derive the latent semantic knowledge based on the SVD operation. Similar to finding relevant web pages, with respect to the query page, LSI could also achieve finding the user sessions exhibiting similar navigational aims to current user's navigational activity efficiently [15]. However, conventional LSI-based web usage mining approaches are lack of capturing semantic space associated with the discovered web objects. In the following part of this paper, we aim to deal with web usage mining for discovering web usage pattern by combining latent semantic analysis and web clustering.

3.1. Usage Data Identification

Basically, according to W3C definition, a web pageview can be viewed as a visual rendering of a web page. In this way, the user access interest exhibited may be reflected by the varying degree of visits in different web pages during one session. Thus, we can represent a user session as a collection of transactions, which includes a series of weighted pageviews, during the visiting period. In other words, the user session can be expressed in the form of pageview vectors. From such viewing point, we generate the following user session expression. Given n web pages in a web site and m web users visiting the web site during a period of time, after appropriate data preprocessing such as page identification and user sessionization, we built up the pageview corpus as $P=\{p_1,p_2,...p_n\}$, and user session collection as $S=\{s_1,s_2,\cdots,s_m\}$. In short, each user session can be expressed as a set of weight-page pairs, $s_i=\{<p_1,a_{i1}>,<p_2,a_{i2}>,...<p_n,a_{in}>\}$. By simplifying the above expression in the form of pageview vector, each user session can be considered as an n-dimensional vector $s_i=\{a_{i1},a_{i2},...a_{in}\}$, where a_{ij} denotes the weight for pageview p_j in s_i user session. As a result, the whole user session data can be utilized to form web usage data represented by a session-pageview matrix $SP_{m\times n}=\{a_{ij}\}$ (Figure 2 illustrates the skeletal structure of session-page matrix).

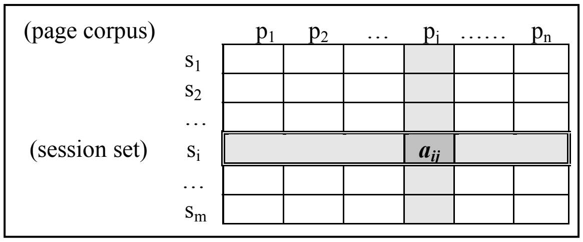


Figure 2. Scheme of session-pageview matrix

The cell value in the session-page matrix, a_{ij}, can be represented by a weight associated with the contribution of page p_j in the user session s_i, which is usually determined by the number of hit or the amount time spent by specific user on the corresponding page. Generally, in order to eliminate the influence caused by the relative amount difference of visiting time duration or hit number, the normalization manipulation across pageviews space in same user session is performed.

3.2. Latent Usage Information

Once usage matrix is constructed, we may applying conventional clustering on user transaction data to classify user sessions into various groups, within which the classified sessions share both common access interest exhibited from their visiting records. It is intuitive to perform clustering algorithm directly on each row vector of usage matrix to determine the relative "close" session cluster by using similarity-based measure, such as commonly adopted cosine similarity from Information Retrieval. In [11], an algorithm named PACT is proposed based on the above discussed technique. However, this kind of clustering technique only capture the mutual relationships between session data explicitly, it is incapable of revealing the "deeper" underlying characteristics of usage pattern. In this work, we propose the *Latent Usage Information* (LUI) algorithm to group user sessions semantically through taking latent information into account. For better understanding the LUI algorithm, we first discuss some theoretical background of the SVD algorithm.

3.2.1. Single Value Decomposition Algorithm

The SVD definition of a matrix is illustrated as follows[16]: For a real matrix $A=\left[a_{ij}\right]_{m\times n}$, without loss of generality, suppose $m\geq n$ and there exists SVD of A:
$A=U_{m\times m}\sum_{m\times n}V_{n\times n}$, where U and V are orthogonal matrices. Matrices U and V can be respectively denoted as $U_{m\times m}=\left[u_1,u_2,\cdots,u_m\right]_{m\times m}$ and $V_{n\times n}=\left[v_1,v_2,\cdots,v_n\right]_{n\times n}$, where $u_i,\ (i=1,\cdots,m)$ is a *m*-

dimensional vector $u_i = (u_{1i}, u_{2i}, \cdots, u_{mi})^T$ and v_j, $(j = 1, \cdots, n)$ is a n-dimensional matrix $v_j = (v_{1j}, v_{2j}, \ldots, v_{nj})^T$. Suppose $rank(A) = r$ and single values of A are diagonal elements of Σ as follows: $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_n = 0$ ·

For a given threshold ε $(0 < \varepsilon \leq 1)$, we choose a parameter k such that $(\sigma_k - \sigma_{k+1}) / \sigma_k \geq \varepsilon$. Then, we denote $U_k = [u_1, u_2, \ldots, u_k]_{m \times k}$, $V_k = [v_1, v_2, \ldots, v_k]_{n \times \mathrm{k}}$, $\Sigma_k = diag(\sigma_1, \sigma_2, \ldots \sigma_k)$ and $A_k = U_k \Sigma_k V_k$.

As known from the theorem in algebra [16], A_k is the best approximation matrix to A and conveys main and latent information among the usage data. This property makes it possible to find out relative "close" user session at the semantic latent level based on their mutual similarity.

3.2.2. Representation of User Transaction in Latent Space

Once SVD implementation is completed, we may rewrite user sessions with the obtained approximation matrix U_k, Σ_k and V_k and map them into another k-dimensional latent space. For a given session, it is represented as a coordinate vector with respect to pageviews: $s_i = \{a_{i1}, a_{i2}, \cdots, a_{in}\}$. The projection of coordinate vector s_i in the k-dimensional latent subspace is re-parameterized as

$$s_i^{'} = s_i V_k \sum{}_k = (t_{i1}, t_{i2}, \ldots, t_{ik}) \tag{7}$$

where $t_{ij} = \sum_{k=1}^{n} a_{ik} v_{kj} \sigma_j$, $j = 1, 2, \ldots, k$.

3.2.3. Similarity Measure

We adopt traditional Cosine similarity to capture common interests shared by user sessions, i.e. for two vectors $x = (x_1, x_2, \ldots, x_k)$ and $y = (y_1, y_2, \ldots, y_k)$ in k-dimensional space, the similarity between them is defined as $sim(x, y) = (x \cdot y) / (\|x\|_2 \|y\|_2)$, where $x \cdot y = \sum_{i=1}^{k} x_i y_i$, $\|x\|_2 = \sqrt{\sum_{i=1}^{k} x_i^2}$. In this manner, the similarity between two user sessions is defined as:

$$sim(s_i^{'}, s_j^{'}) = \frac{(s_i^{'} \cdot s_j^{'})}{\|s_i^{'}\|_2 \, \|s_j^{'}\|_2} \tag{8}$$

3.3. Clustering User Sessions based on Latent Usage Information

In this section, we present the algorithms for clustering user sessions and generating user profile based on the discovered clusters as well.

Here we adopt a modified standard K-means clustering algorithm to classify user sessions based on the transformed SP matrix over the latent k-dimensional space and derive the centroid of cluster obtained as user profiles to represent various user access patterns.. The algorithm is described as follows:

[Algorithm]: Clustering user sessions for user profiles
[Input]: usage data $SP^{'}$ and a similarity threshold ε
[Output]: user session clusters and user profiles

Step 1: Choose the first user session $s_i^{'}$ as the initial cluster C_1 and centroid of this cluster, i.e. $C_1 = \{s_i^{'}\}$ and $Cid_1 = s_i^{'}$.

Step 2: For each session $s_i^{'}$, calculate the similarity between $s_i^{'}$ and the centroids of other existing cluster $sim(s_i^{i}, Cid_j)$.

Step 3: if $sim(s_i^{'}, Cid_k) = \max_j(sim(s_i^{'}, Cid_j)) > \varepsilon$, then allocate $s_i^{'}$ into C_k and recalculate the centroid of cluster C_k as $Cid_k = 1/|C_k| \sum_{j \in C_k} s_j^{'}$;

Step 4: Otherwise, let $s_i^{'}$ itself construct a new cluster and be the centroid of this cluster.

Step 5: Repeat step 2 to 4 until all user sessions are processed and all centroids do not update any more.

Step 6: For each page in clusters, we compute the mean value of pageview as

$$wt(p, pf) = 1/|C_k| \sum_{s \in C_k} w(p, s) \tag{9}$$

where $w(p,s)$ is the weight of the page p in session s, $s \in C_k$

Step 7: For each page weight smaller than a threshold μ, the corresponding item will be removed, otherwise keep it. Sort the pages with their weights in a descending order and output the mean vectors as user profiles.

$pf_{c_k} = \{< p_{1k}, wt(p_{1k}, pf) >, < p_{2k}, wt(p_{2k}, pf) > \ldots, < p_{tk}, wt(p_{tk}, pf)\}$

where $wt(p_{1k}, pf) > wt(p_{2k}, pf) > \cdots > wt(p_{tk}, pf) > \mu$, $PF = \{pf_{c_k}\}, k = 1, 2 \cdots, t$.

3.4. Experiment and Evaluation

In order to evaluate the effectiveness of the proposed LUI-based clustering algorithm and user profile generating algorithm, and explore the discovered user access pattern, we conducted experiments on two real world data sets.

3.4.1 Data Sets

The first dataset used is downloaded from KDDCUP (www.ecn.purdue.edu/ kddcup/). Data preprocessing is needed to perform on the raw data set since there are some short user sessions existing in the data set, which mean they are of less contribution for data mining. After data preparation, we have setup a data set including 9308 user sessions and 69 pages, where every session consists of 11.88 pages in average. We refer this data set to "KDDCUP data". In this data set, the entries in session-page matrix associated with the specific page in the given session are determined by the numbers of web page hits by the given user.

The second data set is from a university website log files and was made available by the author of [17]. The data is based on a random collection of users visiting this site for a 2-week period during April of 2002. After data preprocessing, the filtered data contains 13745 sessions and 683 pages. This data file is expressed as a session-page matrix where each column is a page and each row is a session represented as a vector. The entries in the table correspond to the amount of time (in seconds) spent on pages during a given session. For convenience, we refer this data as "CTI data". For each dataset, we randomly choose 1000 transaction as the evaluation set, whereas the remainder part is selected as the training set for constructing user profiles.

3.4.2. Results of Generated User Profiles

We utilize aforementioned LUI method to classify user sessions. From the results, it is found that generated profiles are "overlapping" of pageviews since some pageviews are listed in more than one user clusters. Table 2 depicts 3 user profiles generated from KDD dataset using LUI approach. Each user profile is listed in a ordered pageviews' sequence with weights, which means the greater weight of a pageview contribute, the more likely it is to be visited. The first profile in Table 2 represents the activities involved in online-shopping circumstance such as login, shopping_cart, checkout etc., especially occurring in purchasing leg-wear products, whereas second user profile reflects customers' concern focused on the interests with regard to the department store itself.

Analogously, some informative finding can be obtained in Table 3, which is derived from CTI dataset. In this table, three profiles are generated: the first one reflects the main topic of international student concerning issues regarding applying for admission, and second one involves in the online applying process for graduation, whereas the final one indicates the most common activities happened during students browsing the university website, especially while they are determining course selection, i.e. selecting course, searching syllabus list, and then going through specific syllabus.

Table 2. Examples of user profiles from KDD dataset

Pageview #	Pageview content	weight
29	Main-shopping_cart	1.00
4	Products-productDetailleagwear	0.86
27	Main-Login2	0.67
8	Main-home	0.53
44	Check-expressCheckout	0.38
65	Main-welcome	0.33
32	Main-registration	0.32
45	Checkout-confirm_order	0.26

Pageview #	Pageview content	weight
11	Main-vendor2	1.00
8	Main-home	0.40
12	Articles-dpt_about	0.34
13	Articles-dpt_about_mgmtteam	0.15
14	Articles-dpt_about_broadofdirectors	0.11

Table 3. Examples of user profiles from CTI dataset

Pageview #	Pageview content	weight
19	Admissions-requirement	1.00
3	Admissions-costs	0.41
15	Admissions-intrnational	0.24
13	Admissions-I20visa	0.21
387	Homepage	0.11
0	Admission	0.11

Pageview #	Pageview content	weight
349	Gradapp-tologin	1.00
20	Admissions-statuscheck	0.35
340	Gradapp-login	0.32
333	Gradapp-appstat_shell	0.13
0	Admissions	0.11

Pageview #	Pageview content	weight
387	Homepage	1.00
59	Courses	0.78
71	Course-syllabilist	0.40
661	Program-course	0.17
72	Course-syllabisearch	0.12

4. Using Tensor Analysis for Finding Co-Clusters of Web Objects.

In this section, we introduce a recently emerging high-order data analysis method, a so-called tensor-based latent analytic algorithm, to address finding co-clusters of web objects in web usage mining. Unlike the usage data model used in Section 3, here, a user session adjacency tensor is formed by treating the visited web pages as a third dimension and incorporating it into calculation of association degrees with the latent factors. This algorithm is executed via a three-way Parallel Factors (PARAFAC) decomposition. As a result of the latent analysis capability, it is considered as a high-order analogue of the Singular Value Decomposition (SVD), which is described in the Latent Usage Information (LUI) algorithm in Section 3 and widely used in other Latent Semantic Indexing (LSI) algorithms in many applications of text retrieval.

4.1. Notations

To better describe the proposed algorithm, we first introduce a set of notations used in following discussion.

- Scalars are denoted by lowercase letter, *a*.
- Vectors are denoted by boldface lowercase letters, **a**. The *i*-th entry of **a** is denoted by $\mathbf{a}_i$.
- Matrices are denoted by boldface capital letters, e.g., **A**. The j-th column of **A** is donated by $\mathbf{a}_j$ and element by a_{ij}
- Tensors, multi-way arrays, are denoted by boldface Euler script letters, e.g., $\boldsymbol{\mathcal{X}}$. Element (*i,j,k*) of a 3rd-order tensor $\boldsymbol{\mathcal{X}}$ is denoted by x_{ijk}.
- The symbol ○ denotes the outer product of vectors; for example, if $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^J$, $\mathbf{c} \in \mathbb{R}^K$, then $\boldsymbol{\mathcal{X}}=\mathbf{a}\circ\mathbf{b}\circ\mathbf{c}$ if and only if $x_{ijk} = a_i b_j c_k$ for all $1 \le i \le I$, $1 \le j \le J$, $1 \le k \le K$.
- The notation $X\ \bar{x}_i\ a$ indicates that the tensor $\boldsymbol{\mathcal{X}}$ should be multiplied by the vector **a** in dimension i▷ψForψexample‹ψ $h = X\ \bar{x}_2\ a\ \bar{x}_3\ b$ means to multiply $\boldsymbol{\mathcal{X}}$ by vector **a** in the 2nd dimension and vector **b** in the 3rd dimension

4.2. TOPCLUS Algorithm

In [18], authors proposed an algorithm, called TOPHITS method, which could be considered as an extension of the traditional HITS algorithm to linkage analysis in a three-way representation of web hyperlink structure. We aimed to introduce this algorithm into web usage mining, which is to discover the co-clusters of web objects via a three-way tensor decomposition. The aims of the algorithm operation are not only to discover the significant web user sessions and web pages, which are closely related to the user access topics, but also identify the topic-based aggregations of web objects (i.e. web user sessions and pages). In this case, we called this algorithm as topic-based co-clustering (TOPCLUS) algorithm.

Let *K* be the number of pages that web users have visited, *N* be the number of user sessions in web log files. In TOPCLUS, the $N \times N \times K$ adjacency tensor is defined as

$$x_{ijk} = \begin{cases} \text{m, the total number of visits on page k by session i and j} \\ \text{0, otherwsie} \end{cases}$$

$$\text{for } 1 \le i, j \le N,\ 1 \le k \le K$$

The TOPCLUS algorithm uses the PARAFAC model to generate a rank-R approximation of the original tensor expression

$$X \approx \lambda [\![A,B,T]\!] \equiv \sum_{r=1}^{R} \lambda^{(r)} a^{(r)} \circ b^{(r)} \circ t^{(r)} \qquad (9)$$

Here we assume that $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_R$. The PARAFAC decomposition approximates the tensor $\boldsymbol{\mathcal{X}}$ by the sum of R rank-1 outer products shown in figure 3. In this manner, the PARAFAC decomposition is a three-order analogue of SVD.

To perform PARAFAC decomposition, a so-called greedy PARAFAC algorithm is used [18]

```
[Algorithm]: Greedy PARAFAC Decomposition
[Input]: $\boldsymbol{\mathcal{X}}$ ={ $x_{ijk}$ } of tensor in size of $N \times N \times K$
[Output]: A rank-R approximation of $\boldsymbol{\mathcal{X}}$, returned as R triplet {$\mathbf{a}^{(r)}$, $\mathbf{b}^{(r)}$, $\mathbf{t}^{(r)}$} plus weight $\lambda^{(r)}$, where $r = 1,2,\cdots,R$
For $l = 1,2,\cdots,R$, do:
    Initialize x, y, z to be vectors of all ones of length N, N, and K, respectively
    Repeat:
        $x = X \bar{\times}_2 y \bar{\times}_3 z - \sum_{r=1}^{l-1} \lambda^{(r)} a^{(r)} (y^T b^{(i)})(z^T t^{(i)})$
        $y = X \bar{\times}_1 x \bar{\times}_3 z - \sum_{r=1}^{l-1} \lambda^{(r)} b^{(r)} (x^T a^{(i)})(z^T t^{(i)})$
        $z = X \bar{\times}_1 x \bar{\times}_2 y - \sum_{r=1}^{l-1} \lambda^{(r)} t^{(r)} (x^T a^{(i)})(y^T b^{(i)})$
        $\delta = \|x\|_2 \|y\|_2 \|z\|_2$, and normalize x, y, z until
       the change in $\delta$ is small
    Set $a^{(l)} = x$, $b^{(l)} = y$, $t^{(l)} = z$ ,and $\lambda^{(l)} = \delta$
  End do
```

After performing the Greedy PARAFAC, we obtain the estimates of $a^{(r)}$, $b^{(r)}$ and $t^{(r)}$, which represent the contributions of user sessions and pages on various topics/factors.

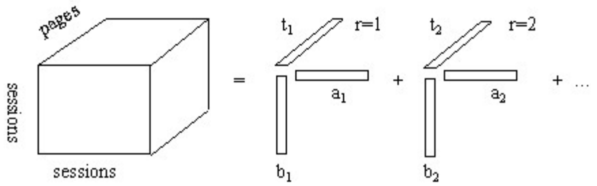
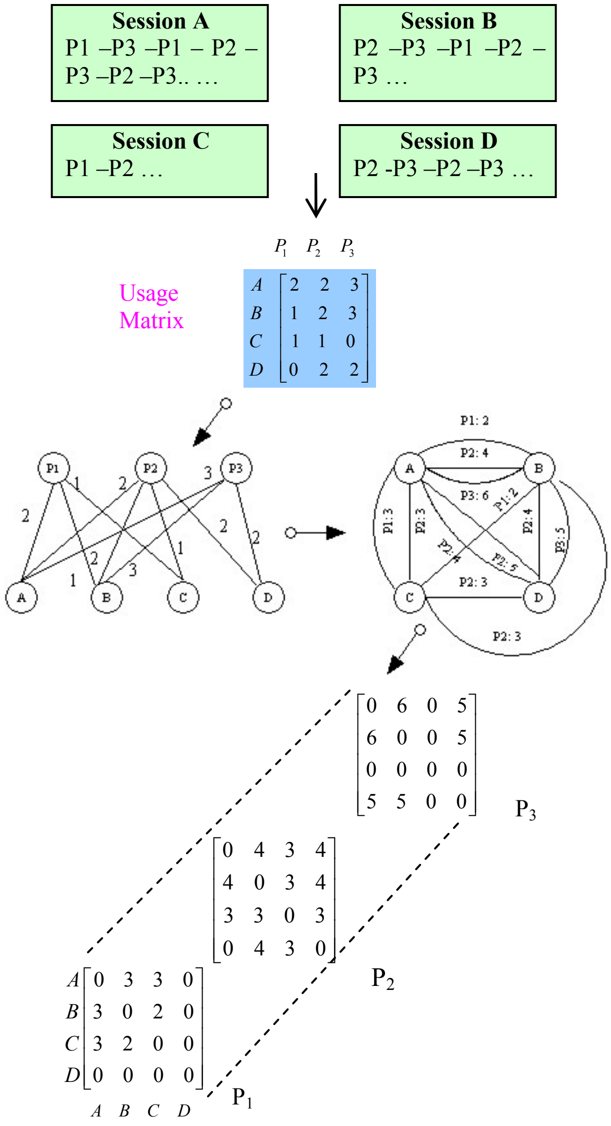


Figure 3. The illustration of PARAFAC decomposition in a three-way adjacency tensor

4.3. Using Tensor Decomposition for Finding Co-Clusters of Web Objects

In this section, we aimed to re-model the usage data derived web log files in a three-way tensor expression. We started from the session-page usage matrix described in Section 3. Based on the defined usage matrix, we can first build up a bipartite network of web objects, where the upper and lower row nodes denote the visited pages and user sessions, respectively, to characterize the web access links between user sessions and web pages from the web log files. Upon the bipartite network, we further construct a labeled network of user sessions to reflect the common interest of various user sessions. In this network, nodes represent the user sessions, the line labeled with individual web page between two user sessions denotes the common interest on the corresponding page, and the weight is determined by the total number of visits by these two user sessions. Figure 4 depicts a process of transforming an example of web log file snapshots into a three-way tensor expression.

Upon the constructed usage expressions in a three-way tensor space, we employ the PARAFAC decomposition operation described above to calculate the scores of $a^{(r)}$ $b^{(r)}$, $t^{(r)}$ on various topics, $1 \le r \le R$. Since these scores reflect the association of web objects with the latent topics, the web objects with scores exceeding a certain value will be considered as the significant members of co-clusters of web objects that significantly contribute to the corresponding topics. In the other words, the co-clusters of web objects in the forms of either page-term or session-page aggregations mainly reflect the intrinsic linking and aggregating property of web data. More experiment investigations for finding the significant pages and the associated topics are found in [18, 19].


Session A
P1 –P3 –P1 – P2 – P3 –P2 –P3.. …

Session B
P2 –P3 –P1 –P2 – P3 …

Session C
P1 –P2 …

Session D
P2 -P3 –P2 –P3 …

Usage Matrix

$$\begin{array}{c|ccc} & P_1 & P_2 & P_3 \\ \hline A & 2 & 2 & 3 \\ B & 1 & 2 & 3 \\ C & 1 & 1 & 0 \\ D & 0 & 2 & 2 \end{array}$$

$$P_3: \begin{bmatrix} 0 & 6 & 0 & 5 \\ 6 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 \\ 5 & 5 & 0 & 0 \end{bmatrix}$$

$$P_2: \begin{bmatrix} 0 & 4 & 3 & 4 \\ 4 & 0 & 3 & 4 \\ 3 & 3 & 0 & 3 \\ 0 & 4 & 3 & 0 \end{bmatrix}$$

$$P_1: \begin{array}{c|cccc} & A & B & C & D \\ \hline A & 0 & 3 & 3 & 0 \\ B & 3 & 0 & 2 & 0 \\ C & 3 & 2 & 0 & 0 \\ D & 0 & 0 & 0 & 0 \end{array}$$

Figure 4. The construction of a tensor expression from an example of web log file

4.4. Web Community Analysis and Social Network Analysis

As we discussed, web community analysis is to discover the aggregations of web pages, users as well as co-clusters of web objects. As a result, web communities are always modeled as groups of pages and users, which can also be represented by various graphic expressions, for example, here the nodes denote the users, while the lines stand for the relationships between two users, such as pages commonly visited these two users or email communications between senders and receivers. In the other words, a web community is modeled as a network of users exchanging information or exhibiting common interest, that is, a social network. In this sense, the gap between web community analysis and social network analysis becomes closer and closer, many concepts and techniques used and developed in one area could be extended into the research area of the other.

On the other hand, with the prevalence and maturity of web 2.0 technologies, the web is becoming a useful platform and an influential source for individuals to share their information and express their opinions. For example, blogs space like *myspace* and *facebook* is becoming a global information sharing and exchanging source. From this viewing point, how to extend the current web community analysis to a very big scale data source or how to introduce the achievements from traditional social network analysis into web data managements is emerging a huge amount of challenges that web researchers and engineers have to face.

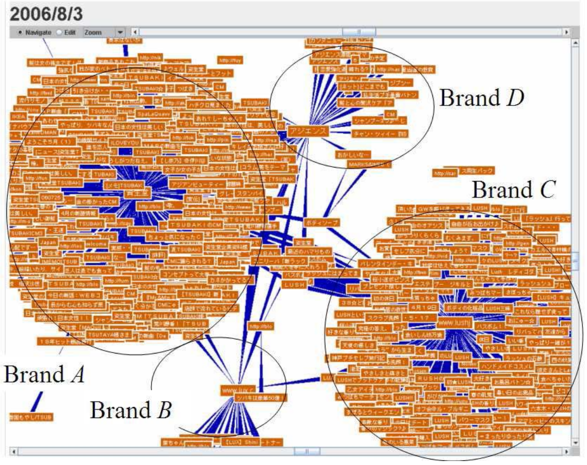


Figure 5. Comparison of commodity brands

Recently some research studies on web community analysis have addressed the investigation of societal behavior evolution on the Web. In [20], Kitsuregawa et al. used a structural and temporal analytic system to investigate the societal development of blogs space. In this study, a blog piece was considered a specific web page, the inter-blog links are treated as the equivalent to the web link. By examining the structural and temporal developments of the web blog community, we can find many interesting and unseen outcomes. Figure 5 illustrates an application of this analysis to comparing the commodity brand names in human minds.

In another application, [21] proposed a new algorithm *ASALSAN for computing three-way DEDICOM* (DEcomposition into DIrectional COMponents), which was to calculate latent components in data and the pattern of asymmetric (i.e., directed) relationships among these components. In two case studies of trade among nations or the exchange of emails among individuals, authors aimed to analyze the intrinsically asymmetric relationships by incorporating a third mode of the data, i.e. time. The study started on modeling a set of interaction activities as a social network graph and discretizing the analyzed dataset into a three-way tensor representation over a 10-years time span. Then the proposed algorithm was employed to decompose the three-way original space and extract a concise semantic graph, which consists of a number of latent components. From the calculated components, the main interactive attributes are visualized as a directed bipartite graph. Moreover, the developed algorithm was also able to capture the evolution of the temporal patterns of the semantic graph over time. It is believed that much work on this topic will be seen in near future.

5. Conclusion and Future Work

In this paper, we have investigated using web clustering for web community mining and analysis. In the context of web linkage analysis, we proposed a novel correlation based web similarity and web clustering algorithm for finding the linking-relevant page groups via linkage structure analysis. The direct and indirect (via correlation measure) linkage knowledge was analyzed to reveal the web page communities. In the case of web usage mining, we have combined the latent semantic analysis and web clustering to identify user session aggregations in the form of web access patterns. Experimental results have demonstrated the proposed method could reveal web access patterns in a latent semantic space explicitly. In addition to clustering web pages or user sessions alone, we also investigated using a three-way tensor expression for modeling a web access observation in a unified high-order space and capturing the co-aggregations of web objects via a three-way latent semantic approximation decomposition. We further addressed the combination of web community analysis with social network analysis to keep pace with the recent emergence and quick developments of web 2.0 based applications

In future work we aim to analyze the impact of various factors such as data scalability and sparsity on web clustering, another direction is how to introduce the concepts and techniques of social network analysis for web community mining and analysis.

6. Reference

1. Zhang, Y., J.X. Yu, and J. Hou, Web Communities: Analysis and Construction. 2006, Berlin Heidelberg: Springer.

2. Kleinberg, J., Authoritative Sources in a Hyperlinked Environment. Journal of the ACM, 1999. 46(5): p. 604-632.

3. Brin, S. and L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. in Proceedings of the 7th International World Wide Web Conference. 1998, Brisbane, Australia.

4. Li, L., Shang Y, and Z. W. Improvement of HITS-based Algorithms on Web Documents. in Proceedings of WWW2002. 2002, Honolulu, Hawaii, USA.

5. Wang, Y. and M. Kitsuregawa. Use Link-based Clustering to Improve Web Search Results. in Proceedings of the 2nd International Conference on Web Information Systems Engineering (WISE2001). 2001, p. 119-128, Kyoto, Japan.

6. Chakrabarti, S., et al., Mining the Web's Link Structure. Computer, 1999. 32(8): p. 60-67.

7. Hou, J. and Y. Zhang, Effectively Finding Relevant Web Pages from Linkage Information. IEEE Trans. Knowl. Data Eng., 2003. 15(4): p. 940-951.

8. Borodin, A., et al. Finding Authorities and Hubs from Hyperlink Structures on the World Wide Web. in Proceedings of the 10th International World Wide Web Conference. 2001, p. 415-429, Hong Kong, China.

9. Brin, S. and L. Page, The PageRank Citation Ranking: Bringing Order to the Web (http://www-db.stanford.edu/~backrub/pageranksub.ps.). 1998.

10. Hou, J. and Y. Zhang. Utilizing Hyperlink Transitivity to Improve Web Page Clustering. in Proceedings of the 14th Australasian Database Conferences (ADC2003). 2003, p. 49-57, Adelaide, Australia: ACS Inc.

11. Mobasher, B., et al., Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization. Dating and Applications (ADMA 2005). 2005, p. 31-42, Wuhan, china: Springer.

16. Datta, B.N., a Mining and Knowledge Discovery, 2002. 6(1): p. 61-82.

12. Perkowitz, M. and O. Etzioni. Adaptive Web Sites: Automatically Synthesizing Web Pages. in Proceedings of the 15th National Conference on Artificial Intelligence. 1998, p. 727-732, Madison, WI: AAAI.

13. Dumais, S.T. Latent semantic indexing (LSI): Trec-3 report in Proceeding of the Text REtrieval Conference (TREC-3). 1995, p. 219-230, Gaithersburg, USA.

14. Deerwester, S., et al., Indexing by latent semantic analysis. Journal American Society for information retrieval, 1990. 41(6): p. 391-407.

15. Xu, G., Y. Zhang, and X. Zhou. A Latent Usage Approach for Clustering Web Transaction and Building User Profile. in The First International Conference on Advanced Data MinNumerical Linear Algebra and Application. 1995: Brooks/Cole Publishing Company.

17. Mobasher, B., Web Usage Mining and Personalization, in Practical Handbook of Internet Computing, M.P. Singh, Editor. 2004, CRC Press. p. 15.1-37.

18. Kolda, T.G., B.W. Bader, and J.P. Kenny. Higher-Order Web Link Analysis Using Multilinear Algebra. in Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005). 2005, p. 242-249, Houston, Texas, USA: IEEE Computer Society.

19. Kolda, T. and B. Bader. The TOPHITS model for higher-order web link analysis. in Workshop on Link Analysis, Counterterrorism and Security at SDM06. 2006.

20. Kitsuregawa, M., et al. Socio-Sense: A system for analysing the societal behavior from long term Web archive. in Proceeding of APWeb 2008 conference. 2008, p. 1-8, Shenyang, China: Springer.

21. Bader, B.W., R.A. Harshman, and T.G. Kolda. Temporal Analysis of Semantic Graphs Using ASALSAN. in Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007). 2007, p. 33-42, Omaha, Nebraska, USA: IEEE Computer Society.

Biography

Yanchun Zhang is a full Professor of Computer Science and Director of Centre for Applied Informatics Research at Victoria University. Dr Zhang obtained a PhD degree in Computer Science from The University of Queensland in 1991. Prof. Zhang' research interests include databases, cooperative transactions management, web information systems, web mining, web services and e-research. He has published over 160 research papers in international journals and conference proceedings including top journals such as ACM Transactions on Computer and Human Interaction (TOCHI), IEEE Transactions on Knowledge and Data Engineering (TKDE), and a dozen of books and journal special issues in the related areas.

Dr. Zhang is a founding editor and editor-in-chief of World Wide Web and the founding editor of Web Information Systems Engineering and Internet Technologies Book Series from Springer. He is Chairman of International Web information Systems Engineering Society (WISE). He is currently a member of Australian Research Council's College of Experts.