Web Mining Techniques for Recommendation and Personalization

Guandong Xu

A Dissertation submitted to
The School of Computer Science & Mathematics
Faculty of Health, Engineering & Science
Victoria University, Australia

For the degree of
Doctor of Philosophy

March 2008

Doctor of Philosophy Dissertation Declaration

“I, Guandong Xu, declare that the PhD thesis entitled “**Web Mining Techniques for Recommendation and Personalization**” is no more than 100,000 words in length including quotes and exclusive of tables, figures, appendices, bibliography, references and footnotes. This thesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma. Except where otherwise indicated, this thesis is my own work”.

Signature: Date:

Abstract

Nowadays Web users are facing the problems of information overload and drowning due to the significant and rapid growth in the amount of information and the number of users. As a result, how to provide Web users with more exactly needed information is becoming a critical issue in web-based information retrieval and Web applications. In this work, we aim to address improving the performance of Web information retrieval and Web presentation through developing and employing Web data mining paradigms.

Web data mining is a process that discovers the intrinsic relationships among Web data, which are expressed in the forms of textual, linkage or usage information, via analysing the features of the Web and web-based data using data mining techniques. Particularly, we concentrate on discovering Web usage pattern via Web usage mining, and then utilize the discovered usage knowledge for presenting Web users with more personalized Web contents, i.e. Web recommendation.

For analysing Web user behaviour, we first establish a mathematical framework, called the usage data analysis model, to characterise the observed co-occurrence of Web log files. In this mathematical model, the relationships between Web users and pages are expressed by a matrix-based usage data schema. On the basis of this data model, we aim to devise algorithms to discover mutual associations between Web pages and user sessions hidden in the collected Web log data, and in turn, to use this kind of knowledge to uncover user access patterns.

To reveal the underlying relationships among Web objects, such as Web pages or user sessions, and find the Web page categories and usage patterns from Web log files, we have proposed three kinds of latent semantic analytical techniques based on three statistical models, namely traditional Latent Semantic Indexing, Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation model. In comparison to conventional Web usage mining approaches, the main strengths of latent semantic based analysis are their capabilities that can not only, capture the mutual correlations hidden in the observed objects explicitly, but also reveal the unseen latent factors/tasks associated with the discovered knowledge implicitly.

In the traditional Latent Semantic Indexing, a specific matrix operation, i.e. Singular Value Decomposition algorithm, is employed on the usage data to discover the Web user behaviour pattern over a transformed latent Web page space, which contains the maximum approximation of the original Web page space. Then, a k-means clustering algorithm is applied to the transformed usage data to partition user sessions. The discovered Web user session group is eventually treated as a user session aggregation, in which all users share like-minded access task or intention. The centroids of the discovered user session clusters are, then, constructed as user profiles.

In addition to intuitive latent semantic analysis, Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation approaches are also introduced into Web usage mining for Web page grouping and usage profiling via a probability inference approach. Meanwhile, the latent task space is captured by interpreting the contents of prominent Web pages, which significantly contribute to the user access preference. In contrast to traditional latent semantic analysis, the latter two approaches are capable of not only revealing the underlying associations between Web pages and users, but also capturing the latent task space, which is corresponding to user navigational patterns and Web site functionality. Experiments are performed to discover user access patterns, reveal the latent task space and evaluate the proposed techniques in terms of quality of clustering.

The discovered user profiles, which are represented by the centroids of the Web user session clusters, are then used to make usage-based collaborative recommendation via a top-N weighted scoring scheme algorithm. In this scheme, the generated user profiles are learned from usage data in an offline stage using above described methods, and are considered as a usage pattern knowledge base. When a new active user session is coming, a matching operation is carried out to find the most matched/closest usage pattern/user profile by measuring the similarity between the active user session and the learned user profiles. The user profile with the largest similarity is selected as the most matched usage profile, which reflects the most similar access interest to the active user session. Then, the pages in the most matched usage profile are ranked in a descending order by examining the normalized page weights, which are corresponding to how likely it is that the pages will be visited in near future. Finally, the top-N pages in the ranked list are recommended to the user as the recommendation pages that are very likely to be visited in the coming period. To evaluate the effectiveness and efficiency of the recommendation, experiments are conducted in terms of the proposed recommendation accuracy metric. The experimental results have demonstrated that the proposed latent semantic analysis models and related algorithms are able to efficiently extract needed usage knowledge and to accurately make Web recommendations.

Data mining techniques have been widely used in many other domains recently due to the powerful capability of non-linear learning from a wide range of data sources. In this study, we also extend the proposed methodologies and technologies to a biomechanical data mining application, namely gait pattern mining. Likewise in the context of Web mining, various clustering-based learning approaches are performed on the constructed gait variable data model, which is expressed as a feature vector of kinematic variables, to discover the subject gait classes. The centroids of the partitioned gait clusters are used to represent different specific walking characteristics. The data analysis on two gait datasets corresponding to various specific populations is carried out to demonstrate the feasibility and applicability of gait pattern mining. The results have shown the discovered gait pattern knowledge can be used as a useful means for human movement research and clinical applications.

Acknowledgements

First of all, I am sincerely pass my gratitude to my principal supervisor, Professor Yanchun Zhang, for his help, guidance and encouragement throughout the course of my doctoral program at Victoria University, and his criticisms and constructive suggestions on the preparation of the dissertation. His patience, insights, research style and the ability to draw research questions from literature have been integral to the success of this work and to my career development as a researcher. Without his professional guidance and help, this work would not have been achieved. I am also grateful to him for providing me with various supports to conduct this study and many invaluable opportunities to let me be involved in many professional activities, which are very beneficial for my future academic career.

Thanks are also presented to my co-supervisor and external co-supervisor, Dr Bailing Zhang and Professor Xiaofang Zhou, for their constant help, discussion, encouragement and many constructive suggestions throughout my doctoral study, especially during preparing research papers. I also would like to thank many anonymous reviewers for their critical and valuable comments on our papers, which are the basis of this dissertation.

I am grateful to Victoria University for offering me an Australian Postgraduate Research Award Scholarship, which helps me undertake my PhD study, and the School of Computer Science and Mathematics for giving a casual tutor position to support my study, supplying good services and a friendly laboratory environment, and providing a amount of financial support to travel to several conferences throughout my time here. My gratitude also goes to the Head of the School, Associate Professor Petrio Cerios, the Scholarship Coordinator of Office for Postgraduate Research, Ms. Lesley Birth, the School Conference Coordinator, Dr. Alasdair McAndrew, the School Postgraduate Coordinator, Dr. Gitesh Raikundalia, and all staffs in the School and Faculty, as well as my colleagues in ITArL laboratory for their helps and supports, which provide countless assistance and suggestions.

The last but not the least, I would like to express my gratitude to my wife Feixue and son Jack for their love, support, encouragement, as well as understanding and patience.

Publication Based on This Dissertation

1. Y. Zhang and G. Xu (Correspondence author), On Web Communities Mining and Recommendation, Concurrency and Computation: Practice and Experience, Journal, 2008 (In Press)

2. Y. Zhang and G. Xu, On Web Communities Mining and Analysis, in Proceeding of the 3rd international conference on Semantic, Knowledge and Grid (SKG2007), pp 20-25, Oct 29-31, Xi’an, China, 2007

3. G. Xu, Y. Zhang, R. Begg, Mining gait pattern for clinical locomotion diagnosis based on clustering technique, in Proceedings of the Second International Conference of Advanced Data mining and Applications (ADMA’2006), LNAI 4903, pp 296-307, Xi'An, China, 2006

4. G. Xu, Y. Zhang and X. Zhou, Discovering Task-Oriented Usage Pattern for Web Recommendation, in Proceeding of the 17th Australasian Database Conference (ADC'2006), pp 167-174, January 16 - 19, 2006, Tasmania, Australia, 2006

5. G. Xu, Y. Zhang and X. Zhou, Using Probabilistic Semantic Latent Analysis for Web Page Grouping, in Proceeding of the 15th International Workshop on Research Issues on Data Engineering: Stream Data Mining and Applications (RIDE-SDMA'2005), in conjunction with ICDE'2005, pp 29-36, April 3-4, 2005, Tokyo, Japan.

6. G. Xu, Y. Zhang and X. Zhou, A Web Recommendation Technique Based on Probabilistic Latent Semantic Analysis, in Proceeding of the 6th International Conference of Web Information System Engineering (WISE'2005), LNCS 3806, pp 15-28, November 22-25, 2005, New York City, USA.

7. G. Xu, Y. Zhang and X. Zhou, Towards User Profiling for Web Recommendation, in Proceeding of the 18th Australian Joint Conference on Artificial Intelligence (AI'2005), LNAI 3809, pp 405-414, December 5-9, 2005, Sydney, Australia.

8. Y. Zhang, G. Xu and X. Zhou, A Latent Usage Approach for Clustering Web Transaction and Building User Profile, in Proceeding of the First International Conference on Advanced Data Mining and Applications (ADMA2005), LNAI 3584, pp 31-42, July 22-24, 2005, Wuhan, china.

9. G. Xu, Y. Zhang, J. Ma and X. Zhou, Discovering User Access Pattern Based on Probabilistic Latent Factor Model, in Proceedings of the 16th Australasian Database Conference (ADC 2005), pp 27-36, 31 January - 3 February 2005, Newcastle, Australia.

Table of Content

List of Figures

List of Tables

1. Introduction

1.1. Overview

With the dramatically quick and explosive growth of information available over the Internet, World Wide Web has become a powerful platform to store, disseminate and retrieve information as well as mine useful knowledge. Due to the properties of the huge, diverse, dynamic and unstructured nature of Web data, Web data research has encountered a lot of challenges, such as scalability, multimedia and temporal issues etc. As a result, Web users are always drowning in an "ocean" of information and facing the problem of information overload when interacting with the web. Typically, the following problems are often mentioned in Web related research and applications:

(1). Finding relevant information: To find specific information on the web, users often either browse Web documents directly or use a search engine as a search assistant. When a user utilizes a search engine to locate information, he or she often enters one or several keywords as a query, then the search engine returns a list of ranked pages based on the relevance to the query. However, there are usually two major concerns associated with the query-based Web search [1]. The first problem is low precision, which is caused by a lot of irrelevant pages returned by the search engine. The second problem is low recall, which is due to the lack of capability of indexing all Web pages available on the Internet. This causes the difficulty in locating the unindexed information that is actually relevant.

How to find more relevant pages to the query, thus, is becoming a popular topic in Web data management in last decade [2].

(2). Finding needed information: Most search engines perform in a query-triggered way that is mainly on a basis of one keyword or several keywords entered. Sometimes the results returned by the search engine don't exactly match what a user really needs due to the fact of the existence of the homology. For example, when one user with an information technology background wishes to search information with respect to "Python" programming language, he/she might be presented with information on the creatural python, one kind of snake rather than the programming language, given entering only one "python" word as query. In other words, the semantics of Web data [3] is rarely taken into account in the context of Web search.

(3). Learning useful knowledge: With traditional Web search service, query results relevant to query input are returned to Web users in a ranked list of pages. In some cases, we are interested in not only browsing the returned collection of Web pages, but also extracting potentially useful knowledge out of them (data mining oriented). More interestingly, more studies [4-6] have been conducted on how to utilize the Web as a knowledge base for decision making or knowledge discovery recently.

(4). Recommendation/personalization of information: While a user interacts with the web, there is a wide diversity of user's navigational preference, which results in needing different contents and presentations of information. To improve the Internet service quality and increase the user click rate on a specific website, thus, it is necessary for a Web developer or designer to know what the user really wants to do, predict which pages the user is potentially interested in, and present the customized Web pages to the user by learning user navigational pattern knowledge [3, 7, 8].

The above problems place the existing search engines and other Web applications under significant stress. A variety of efforts have been contributed to deal with these difficulties by developing advanced computational intelligent techniques or algorithms from different research domains, such as database, data mining, machine learning, information retrieval and knowledge management etc. Therefore, the emerging of the Web has put forward a great number of challenges to Web researchers and engineers for web-based data management and Web application development.

Characteristics of Web Data

For the data on the web, it has its own distinctive features compared to the data in conventional database management systems. Web data usually exhibits the following characteristics:

- The data on the Web is huge in amount. Currently, it is hard to estimate the exact data volume available on the Internet due to the exponential growth of Web data every day. For example, in 1994, one of the first Web search engines, the World Wide Web Worm (WWWW) had an index of 110,000 Web pages and Web accessible documents. As of November, 1997, the top search engines claim to index from 2 million (WebCrawler) to 100 million Web documents (from [9]). The enormous volume of data on the Web makes it difficult to handle Web data via well traditional database techniques.
- The data on the Web is distributed and heterogeneous. Due to the essential property of the Web being an interconnection of various nodes over the

Internet, Web data is usually distributed across a wide range of computers or servers, which are located at different places around the world. Meanwhile, Web data is often exhibiting the intrinsic nature of multimedia: that is, in addition to textual information, which is mostly used to express content in terms of text message, many other types of Web data, such as images, audio files and video slips are often included in a Web page. It requires the developed techniques for Web data processing with the ability of dealing with heterogeneity of multimedia data.

- The data on the Web is unstructured. There are, so far, no rigid and uniform data structures or schemas which Web pages should strictly followe, that are common requirements in conventional database management. Instead, Web designers are able to arbitrarily organize related information on the Web together in their own ways, as long as the information arrangement meets the basic layout requirements of Web documents, such as HTML format. Although Web pages in well-defined HTML format could contain some preliminary Web data structures, e.g. tags or anchors, these structural components, however, can primarily benefit the presentation quality of Web documents rather than reveal the semantics contained in Web documents. As a result, there is an increasing requirement to better deal with the unstructured nature of Web documents and extract the mutual relationships hidden in Web data for facilitating users to locate needed Web information or service.
- The data on the Web is dynamic. The implicit and explicit structure of Web data is updated frequently. Especially, due to different applications of web-

based data management systems, a variety of presentations of Web documents will be generated as contents in database updates. And dangling links and relocation problems will be produced when domain or file names changes or disappear. This feature leads to frequent schema modifications of Web documents, which often suffer traditional information retrieval.

The aforementioned features indicate that Web data is a specific type of data different from the data residing in traditional database systems. As a result, there is an increasing demand to develop more advanced techniques to address Web information search and data management. According to the aims and purposes, these studies and developments are mainly about two aspects of Web data management, that is, how to accurately find the needed information on the Internet, i.e. Web information search, and how to efficiently and fully manage and utilize the information/knowledge available from the Internet, i.e. Web data/knowledge management. Especially, with the recent popularity and development of the Internet, such as semantic web, Web 2.0 and so on, more and more advanced Web data based services and applications are emerging for Web users to easily locate the needed information and efficiently share information in a collaborative environment.

Web Data Search

Web search engine technology [10, 11] has emerged catering for the rapid growth and exponential flux of Web data on the Internet, to help Web users find desired information, and has resulted in various commercial Web search engines available online such as *Yahoo!*, *Google*, *AltaVista*, *Baidu* and so on. Search engines can be categorized into two types: one is a general-purpose search engine and another is a specific-purpose search engine. The general-purpose search engines, for example, the well-known Google search engine, are to retrieve as many Web pages available on the Internet that are relevant to the query as possible to Web users. The returned Web pages to user are ranked in a sequence according to their relevant weights to the query, and the satisfaction with the search results from users is dependent on how quickly and how accurately users can find the desired information. The specific–purpose search engines, on the other hand, aim at searching those Web pages for a specific task or an identified community. For example, *Google Scholar* and *DBLP* are two representatives of the specific-purpose search engines. The former is a search engine for searching academic papers or books as well as their citation information for different disciplines, while the latter is designed for a specific researcher community, i.e. computer science, which provides various information regarding conferences or journals in the computer science domain, such as conference homepage, abstracts or full text of papers published in the computer science journals or conference proceedings. *DBLP* has become a helpful and practicable tool for researchers or engineers in computer science area to find the needed literature easily, or to assess the track record of one researcher conveniently. No matter which type the search engine is, each search engine owns a background text database, which is indexed by a set of keywords extracted from the collected documents. To satisfy the higher recall and accuracy rate of the search, Web search engines are requested to provide an efficient and effective mechanism to collect and manage the Web data, and the capabilities to match the user query with the background indexing database quickly and to rank the returned Web contents in an efficient way so that Web user can locate the desired Web pages in a short time or via clicking a few hyperlinks. To achieve these aims, a variety of algorithms or strategies are involved in handling the above mentioned tasks [10-16], which lead to a hot and popular topic in the context of web-based research, i.e. Web data management.

Data mining and its Applications

Data mining is proposed recently as a useful approach in the domain of data engineering and knowledge discovery [17]. Basically, data mining refers to extracting informative knowledge from a large amount of data, which could be expressed in different data types, such as transaction data in e-commerce applications or genetic expressions in bioinformatics research domains. No matter which type of data is, the main purpose of data mining is to discover the hidden or unseen knowledge, normally in the forms of patterns, from the available data repository. Nowadays data mining has attracted more and more attentions from academia and industries, and a great amount of progresses have been achieved in many applications. In the last decade, data mining has been successfully introduced into the research of Web data management, in which a board range of Web objects including Web documents, Web linkage structures, Web user transactions, Web semantics are become the mined targets. Obviously, the informative knowledge mined from various types of Web data can provide us with helps in discovering and understanding the intrinsic relationships among various Web objects, and in turn, will be utilized to benefit the improvement of Web data management [6, 18-23].

Although much work has been done in web-based data management and a great amount of achievement has been made so far, there still remain many open research problems to be solved in this area due to the fact of the distinctive characteristics of Web data, the complexity of Web data model, the diversity of various Web applications, the progress of the related research areas and the increased demands in terms of efficiency and effectiveness from Web users. How to efficiently and effectively handle web-based data management by using more advanced data processing techniques, thus, is becoming an active research topic that is full of many challenges. This is also the main motivation of this study.

1.2. Motivation

Nowadays the Internet has been well known as a big data repository consisting of a variety of data types as well as a large amount of unseen informative knowledge, which can be discovered via a wide range of data mining or machine learning paradigms. All these kinds of techniques are based on intelligent computing approaches, or so-called computational intelligence, which are widely used in the research of database, data mining, machine learning, and information retrieval and so on. Although the progress of the web-based data management research results in developments of many useful Web applications or services, like Web search engines, users are still facing the problems of information overload and drowning due to the significant and rapid growth in the amount of information and the number of users. In particular, Web users usually suffer from the difficulties of finding desirable and accurate information on the Web due to two problems of low precision and low recall caused by the above reasons. For example, if a user wants to search the desired information by utilizing a search engine such as Google, the search engine may provide the user not only the Web content related to the query topic, but also a large mount of irrelevant information. It is sometimes hard for users to obtain their exactly needed information [1, 24] by using conventional search engines alone. Thus, the emerging of Web has put forward a great deal of challenges to Web researchers for web-based information management and retrieval. Web research academia is requested to develop more efficient and effective techniques to satisfy the increasing demands of Web users, such as retrieving the desirable and related information [25], creating good quality Web communities [2, 26], extracting the informative knowledge from the available information [27], capturing the underlying usage pattern from the Web observation data [28], recommending user customized information to offer better Internet service [29], and furthermore mining valuable business information from the common or individual customer navigational behaviour as well [3].

Web (data) mining could be partly used to solve the problems mentioned above directly or indirectly. In principle, Web mining is the means of utilizing data mining methods to induce and extract useful information from Web data information. Web mining research has attracted a variety of academics and engineers from database management, information retrieval, artificial intelligence research areas, especially from data mining, knowledge discovery, and machine learning etc. Basically, Web mining could be classified into three categories based on the mining goals, which determine the part of Web to be mined: *Web content mining*, *Web structure mining*, and *Web usage mining* [28, 30]. Web content mining tries to discover the valuable information from Web contents (i.e. Web documents). Generally, Web content is mainly referred to the textual objects, thus, it is also alternatively termed as text mining sometimes [31]. Web structure mining involves in modelling Web site in terms of linking structures. The mutual linkage information obtained could, in turn, be used to construct Web page community or find relevant pages based on the similarity or relevance between Web pages. A successful application addressing this topic is building Web communities [25, 26, 32-35]. Web usage mining tries to reveal the underlying access patterns from Web transaction or user session data that recorded in Web log files [29, 36]. Generally, Web users are usually performing their interest-driven visits by clicking one or more functional Web objects. They may exhibit different types of access interests associated with their navigational tasks during their surfing periods. Thus, employing data mining techniques on the observed usage data may lead to finding the underlying usage patterns. In addition, capturing Web user access interest or pattern can, not only provide helps for better understanding user navigational behaviour, but also for efficiently improving Web site structure or design. This, furthermore, can be utilized to recommend or predict Web contents tailored and personalized to Web users who can benefit from obtaining more preferred information and reducing waiting time [3, 37].

Web recommendation or personalization could be viewed as a process that recommends the customized Web presentations or predicts the tailored Web contents to Web users according to their specific tastes or preferences. To-date, there are two kinds of approaches commonly used in recommender systems, namely content-based filtering and collaborative filtering systems [38, 39]. Content-based filtering systems such as WebWatcher [40] and client-side agent Letizia [41] usually generate recommendation based on pre-constructed user profiles by measuring the similarity of Web content to these profiles, while collaborative filtering systems make recommendation by referring other users' preference that is closely similar to current one. Recently collaborative filtering has been widely adopted in Web recommendation applications and has achieved great successes as well [42-44]. In addition, Web usage mining has been proposed as an alternative method for not only revealing user access patterns, but also making Web recommendations in the past decade [29]. In the context of Web usage mining, one important goal is to extract the informative knowledge from Web log files and identify underlying user functional interest that leads to common navigational activity. Basically, a user profile is created for representing a specific user navigational pattern based on mining the usage data. Moreover, presenting the desired Web content in a personalized style to user is carried out by matching the current active user session with the discovered usage patterns. With the benefit of the great progresses in data mining research communities, many data mining techniques, such as collaborative filtering based on the *k-Nearest Neighbour* (*kNN*) [42-44], Web user or page clustering [29, 45, 46], association rule mining [47, 48] and sequential pattern mining technique [19] have been adopted in current Web usage mining methods. Consequently, many efforts have been contributed and great achievements have been made in such research fields as Web personalization and recommendation systems [40, 41, 49, 50], Web system improvement [51], Web site modification or redesign [46, 52], and business intelligence and e-commerce [53-55].

In most cases, Web usage mining techniques are principally based on Web page information, such as viewing time and frequency, visiting order, as well as a variety of similarity measures employed in the mining processes [36]. Common interests among Web users can be captured directly from these observation data using different similarity-based criteria [56]. The Web users, in turn, will be able to be classified into different categories depending on their different interests. Although these user categories can be represented as user access patterns explicitly, however, they do not reveal the intrinsic characteristics of Web users' navigational activities, nor can they uncover the underlying and unobservable factors associated with the specific usage pattern extracted from Web transaction histories. For example, such discovered usage patterns provide little knowledge of the underlying reasons why such Web pages and user sessions are grouped together. Therefore, there has been an increasing demand for developing techniques that can extract user navigational patterns and discover latent semantic factors associated [57]. Many applicable commercial recommender systems have been launched to satisfy the needs such as electronic news filtering, meeting scheduling and entertainment recommendation etc [21, 58, 59]. Recently, with the increased attention to semantic Web and ontology, there are emerging a lot of challenges for recommender systems to integrate the semantic knowledge from the domain ontology into the whole stages of a recommendation process. Ontology-based knowledge will be taken into account during data preparation, pattern discovery as well as recommendation stage [8]. An ontology-based application system has been developed at the AIFB [60], which is capable of presenting personalized views on the ontology.

On the other hand, *Latent Semantic Indexing* (LSI) is an approach to capture the latent or hidden semantic relationships among co-occurrence activities [61]. In practical applications, *Singular Value Decomposition* (SVD) or *Principal Component Analysis* (PCA) algorithms are employed to generate a reduced latent semantic space, which is the best approximation of the original input space and reserves the main latent information among the co-occurrence activities [61-63]. LSI has been widely used in information indexing and retrieval applications [62, 64], Web linkage analysis [25, 34] and Web page clustering [65]. Although LSI has achieved great successes in some applications, it still has some shortcomings [66], such as the computational difficulty of the sparsity problem in a co-occurrence matrix, the overfitting problems, the capability of capturing the latent semantic factor space etc. To address these, some studies have extended the standard LSI techniques via introducing various statistical theories. Unlike other data mining or machine learning algorithms, this kind of knowledge learning algorithms utilizes a so-called generative model, which learns the needed knowledge from a large amount of data via iteratively computing the likelihood of the observation until reaching an optimal value. This procedure is, therefore, called a model-generative process. *Probability Latent Semantic Analysis* (PLSA) [66] and *Latent Dirichlet Allocation* (LDA) [67] are two representatives of the generative models, which draw much attention from data mining and machine learning research communities recently. The former one is to estimate the correlation among the observation in terms of probability distribution based on Bayesian rule, while the latter is about capturing the association among the observed objects via computing the posterior Dirichlet distribution. Although these two models are based on different statistical background, however, they are working in a similar probability inference manner. The main outstanding strengths of these techniques are, not only the capability of discovering the underlying association among the co-occurrence observations, but also the power of revealing the latent semantic factor space associated.

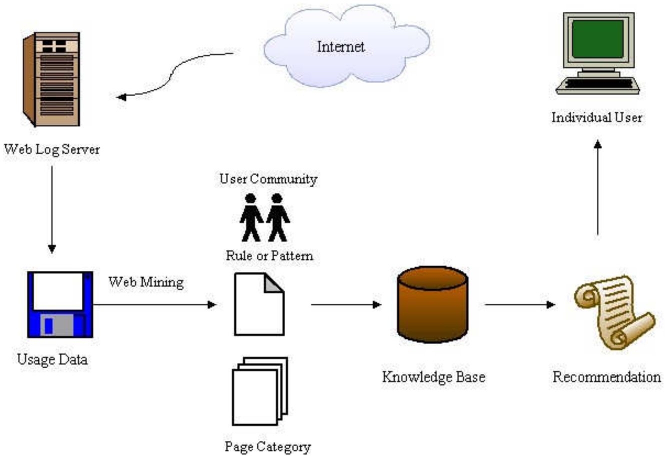


Figure 1-1. The Scheme of Web usage mining and Web recommendation

Both of these approaches discussed are constructed a family of *Latent Semantic Analysis* (LSA) approaches. In this study, we aim to address Web usage mining for Web recommendation by using LSA paradigms. The whole procedure of using Web usage mining for Web recommendation consists of three steps, i.e. data collection and pre-processing, pattern mining (or knowledge discovery) as well as knowledge application. Figure 1-1 depicts the scheme of the process.

To achieve the proposed aims, the objectives of this research are accordingly stated as the following sub-goals:

1. To establish a mathematical framework for Web recommendation based on Web usage mining. For the purpose of extracting latent semantic information conveyed by the Web usage data, it is important to model Web co-occurrence observations within a proper framework, such that the usage information can be analysed and intrinsic associations among Web objects, i.e. Web pages or user sessions, can be captured by employing data mining or machine learning algorithms on the framework. In this work, particularly, we will model the user behaviour as a unified usage data matrix, and exploit the linear algebra, statistical, probability theories and matrix operations to analyse the usage data, discover the underlying relationships among Web objects and predict the Web user navigational preferences. With the introduction of this framework, mining Web usage pattern for Web recommendation can be performed on a solid mathematical base.
2. To discover latent semantic associations among Web objects via the LSI approach. Web usage data usually contains user behaviour information; therefore, the bigger the size of the usage data is, the more informative and comprehensive the usage pattern

knowledge is. In this manner, Web usage data is expressed in a high-dimensional space, which is in a size of hundred thousands or millions of user sessions each being over hundreds to thousands attributes. The high dimension of the original input space usually leads to highly computational difficulties and serious scalability problems, thus, it is necessary to reduce the dimension of the input space without loss of main informative knowledge conveyed by the Web usage data before applying data mining algorithms. LSI algorithms can effectively handle the task of the dimension reduction and latent semantic analysis in Web usage mining. In this work, we first introduce a standard LSI algorithm based on SVD implementation to reveal the intrinsic relationships among the Web objects. Due to the fact that the transformed usage vector space keeps the maximum approximation of the original input space but in a lower dimension, we then intend to measure the similarity of user sessions in the transformed space, and cluster the corresponding user sessions into a number of session clusters, which represent various usage patterns. We propose a LSI-based algorithm to accomplish the above goals. The main advantage of the LSI-based approach is capturing the intrinsic associations among user sessions in a transformed low-dimensional space, which reflects the latent semantic nature of the original input space.

3. To reveal the latent semantic factor space and discover Web user navigational behaviours via the PLSA model. Although the traditional LSI-based knowledge discovery algorithms are able to efficiently reduce the dimensionality of the original input space to better deal with the computing of the high-dimensional vector space and maintaining the major informative knowledge hidden in the observed co-occurrence data, however, they are lack of the capability of identifying the latent semantic factor

space associated with the discovered usage knowledge. In this work, we aim to address this by introducing a variant of the traditional LSI algorithms, a so-called PLSA model, to characterize Web co-occurrence activities and find associations among the Web objects based on Bayesian rule. The PLSA model is theoretically based on an assumption that there is a statistical model, called the aspect model that characterizes the co-occurrence observations; each factor in the aspect model is associated with the co-occurrence activity by a varying degree, which is determined by a conditional probability of Web objects over the factor space. The conditional probability distribution is estimated by an iterative *Expectation-Maximization* (EM) execution, which maximizes the likelihood of the observation data. The derived conditional probability distribution is, in turn, to represent the associations among the Web objects via a probability inference algorithm.

4. To group Web pages based on their usage-oriented similarity. In Web hyperlink analysis, Web page similarity or relevance is usually defined by the linkage information or other related measures like correlation. In this case, the linkage-based page similarity actually reflects the Web page distance in terms of topological information from the viewing point of a Web designer or developer. However, Web users might have different opinions of the relevance of the Web pages while they are visiting a website. Hence the similarity of the Web page based on user navigational behaviour can, sometime, provides an additional means to find the functional closeness of the Web pages, which leads to the improvement of the website structure design and increasing user click rates on a website. Based on this idea, we propose a usage-based Web page similarity to partition Web pages within a specific site into various page

groups. The discovered page groups could be considered as different functional page aggregations.

5. To predict Web user's task preference distributions for Web recommendations. The ultimate goal of Web usage discovery is making use of the usage pattern knowledge for Web recommendations. Because the conditional probability estimates over the latent semantic factor space derived from the PLSA model can be used to predict Web user task preferences, we then make Web recommendations by referring to task-based Web page groups.
6. To recommend the customized Web content by using a user profiling approach. In addition to capturing the Web user's task preference distribution for Web recommendations, another efficient and effective algorithm for Web recommendations is the user profiling approach, which is on a basis of collaborative filtering techniques, a kind of commonly used algorithms in recommender systems. This algorithm works as follows. First we define a new similarity of Web user sessions by introducing the conditional probability of the user session over the latent semantic space. Based on the proposed similarity, the user sessions, which exhibit similar navigational behaviour are partitioned into the same session cluster. The centroid of the discovered user session cluster is viewed as a user profile, which can be considered the representative of one specific usage pattern.
7. To better address Web recommendations by exploiting LDA model. Apart from the PLSA model, LDA model is another generative model, which is based on the computation of posterior probability and Dirichlet value of co-occurrence observations. With LDA model, the associations between Web pages and latent semantic factors,

user sessions and latent semantic factors are modelled by estimating the posterior probability distribution and Dirichlet values via a variant of EM algorithm. We then build up the usage pattern knowledge based on the discovered associations, and make collaborative Web recommendations.

8. Two case studies of the extension of the current study in biomedical data mining domains. The novel user profiling approach used in this study could be extended to develop algorithms for other data mining applications like biomedical data mining applications. In this study, we also adopt the clustering-based user profiling approach in gait pattern mining, which is one important and interesting topic in healthcare and biomechanics domains. We conduct two case studies on two gait datasets, within which one is to investigate the existence of gait patterns in *Cerebral Palsy* (CP) patients and another is for monitoring the fall risk in an elderly population due to ageing or walking impairments.

1.3. Claims of the Dissertation

This dissertation is mainly focused on discovering Web usage patterns in terms of user profiles and latent semantic factors from Web log files to support Web recommendations based various *Latent Semantic Analysis* (LSA) models, and extending the proposed methodologies and technologies to other data mining applications.

The basic research philosophy of this study consists of three procedures: the first is to create a unified mathematical analysis model, on which various data mining algorithms could be employed no matter which research background is addressed. To conduct the Web usage mining, three latent semantic analysis algorithms are investigated and implemented to find Web user groups and Web page categories as well as the latent semantic factors associated, which are used to construct a usage-base Web navigational base via a user profiling method. At the third stage, the discovered usage knowledge is used for a further Web application, i.e. Web recommendation. In addition to Web data mining, the proposed analytical methodology could also be extended to other data mining applications, which use similar research techniques and analytical algorithms but in different application background. In this study, we extend our developed methodologies and technologies to a healthcare data mining domain. This thesis consists of all these studies and results mentioned above. The research contained in this thesis, indeed, spans a number of different research communities – Web mining, Web recommendation, clustering, text retrieval and health data mining, however, it is encircling one focus of using intelligent computational algorithms to deal with knowledge discovery tasks. It is believed it is the main contribution of this thesis.

In particular, to conduct these studies, a mathematical framework is established for Web usage mining and a series of algorithms are proposed to predict Web user navigational preferences and recommend the customized Web contents to Web users. Three kinds of latent semantic analysis models, namely standard LSI, PLSA and LDA, are proposed to address the Web usage mining and Web recommendations respectively. Two case studies of the extension of the proposed pattern mining methodologies and algorithms are carried out in an application of gait pattern mining, one important topic in healthcare and biomechanical data mining domains. The main contributions of the dissertation can be summarized as follows:

A Mathematical Framework of Web Usage Mining for Web Recommendation

This model is based on the matrix theory in linear algebra. Different from other Web data models, such as Web content or Web linkage information, this framework represents the mathematical expression with respect to Web usage information e.g. click number or visiting duration on various Web pages. On the other hand, unlike other usage data models such as directed graphs or visit sequences [19, 68-70], this usage data model is in the form of a usage matrix. From the usage matrix, the intrinsic association among Web objects such as Web pages or user sessions as well as the latent semantic relationships between the latent task space and Web objects conveyed by the usage information, is discovered by a series of matrix operations or generative procedures, such as singular value decomposition, matrix approximation, Bayesian equation conversion, Bayesian updating, Expectation-Maximization iterative operation and so on. Other usage-based information or expressions, such as task-based Web page similarity, task-based user session similarity, user task preference distribution, user profile, and top-N recommended page list are also represented and derived by various matrix operations and other engaged algorithms. This framework makes it feasible to systematically perform analysis on the collected Web usage data using the mathematical theories in a unified way that can extend the developed methodologies and technologies to other data mining applications, which have similar data expressions. As a result, this framework provides a solid mathematical base for discovering Web usage pattern and making Web recommendations.

Latent Semantic Analysis Models For Web Usage Mining In this work, we aim to intensively investigate using latent semantic analysis paradigms for discovering Web usage pattern and making Web recommendations, which includes the following analytical models:

- *Traditional Latent Semantic Indexing* (LSI)
- *Probabilistic Latent Semantic Analysis* (PLSA)
- *Latent Dirichlet Allocation* (LDA)

Tradition LSI is based on a *Singular Value Decomposition* (SVD) operation, which is to reduce the dimensionality of the original input space but holding the maximum approximation of the original matrix. The main advantage of LSI is its capability of uncovering the underlying relationships among the observed objects that aren't exhibited explicitly and directly. In this study, we aim to employ the traditional LSI analysis on the usage data matrix to analyse the associations among user sessions in the transformed vector space resulted from the SVD implementation.

PLSA model is a variant of the tradition LSI models, which introduces an aspect space as an inter-medium between two usage attributes, i.e. user session and Web page. With the PLSA model, the original usage data is mapped into two new usage vectors, in which the associations between user sessions and the latent aspect space, and between Web pages and the latent aspect space, are modelled by the estimates of the conditional probabilities. The new mapped usage vectors along with the newly defined user session similarity and Web page similarity provide a novel Web usage mining method, with which we can derive usage based page groups and session aggregates.

LDA model is a recently emerging generative model, which reveals the intrinsic correlation among co-occurrence via a generative procedure. Different from mining Web usage pattern by the PLSA model, LDA is to learn the hidden usage knowledge based on computing the Dirichlet value and posterior probability. The discovered usage knowledge is then used to predict user potentially interested Web contents. The common strength of the latter two models is the capability of capturing the aspect space that associates with the discovered usage knowledge in addition to usage pattern mining itself.

Algorithms for Web Usage Mining and Web Recommendation In this research, we propose several algorithms and concepts based on three latent semantic analysis models described above respectively.

For the tradition LSI model, the following algorithms and definitions are proposed:

- *Latent Usage Information* (LUI) algorithm. This algorithm is about transforming the original usage data matrix into a latent usage information space, which not only maintains the main usage information within a new dimensionality-reduced usage space, but also reveals semantic relationships hidden in the usage data. In this algorithm, a SVD operation is performed to conduct the latent semantic analysis explicitly.
- User session distance in the semantic space. This measure is to calculate the distance between two user sessions in the semantic space. Due to the advantage of the SVD operation, this kind of distance function is based on a low-dimensional but semantic-based vector space, thus, it is possible to partition user sessions at a level of semantic analysis, that is, the user sessions in same aggregation are more like-minded.

For the PLSA model, the following algorithms and definitions are proposed:

- *Expectation-Maximum* (EM) algorithm. EM algorithm is an iterative operation, which is to estimate the maximum likelihood value of the co-occurrence observations. In the proposed EM algorithm for the PLSA model, we first formulate a set of equations that compute the conditional probability distribution of Web objects against the latent factor space based on Bayesian equation. The EM algorithm starts from an initial input, iteratively executing the Expectation step, which updates the conditional probability distribution, and Maximum step, which aims to re-calculate the likelihood value with the updated conditional probability distribution until reaching a local optimal point. The conditional probability distributions corresponding to the optimal value could be viewed as the final estimates of the relationships between the Web object and the latent factor.

- Usage-based Web page similarity and user session similarity measures. Based on the derived probability estimates via the EM algorithm, we propose two new similarity measures for Web pages and user sessions respectively; one is used for modelling the common functionality of Web pages and another is about measuring the like-minded navigational preferences of user sessions. With the two proposed similarity measures, we are able to find the aggregations of the Web objects by utilizing the discovered usage knowledge.
- Usage-based Web page grouping algorithm. In this study, we develop a new k-means algorithm for grouping Web pages by using the usage-based page similarity. This clustering algorithm generates an automated Web pages groups based on the mutual distance of two pages. The discovered page groups can be, in turn, viewed as the task-driven page aggregations, which can be used to improve or re-structure the Web site design and organization.
- Determining user task preference distribution algorithm. In this algorithm, we aim to identify the user task preference distribution by analysing the very first

clicks of the user via a Bayesian updating approach. The dominant task preferences are determined by selecting those tasks whose corresponding probability weights are exceeding a certain threshold. Incorporating the dominant task preferences with the corresponding tasks characterized by a set of predominant pages, results in the determination of the pages with significant weights as the recommended page list.

- User profiling algorithm for Web recommendation. In this research, we propose a novel user profiling algorithm to represent usage pattern derived from Web usage mining, by using a collaborative filtering approach. The user sessions are first clustered into a number of session clusters based on the usage-based session similarity measure, and the centroids of the discovered user session clusters, in the forms of weighted page sequences, are created as user profiles. When a new active user session is coming, a most matched user profile is selected by measuring the distance between the active user session and the constructed user profiles, and a weighted scoring scheme is then applied to determine the N pages having the top-N highest weights as the page recommendation list. In other words, the recommended pages are chosen by referring to the historic visits by other users, who have the like-minded visiting preferences. In this sense, this algorithm is also called a collaborative recommendation approach.

For LDA model, we develop the following algorithms:

- A variational EM algorithm. In this research, we adopt a variational EM algorithm to find the variational parameters that maximizes the log likelihood of the usage data. The estimates of the variational parameters are the posterior probability and Dirichlet value of the data, the former reflecting the underlying relationships between user sessions and latent factors while the latter representing the linking between Web pages and latent factors.

- Collaborative recommendation algorithm. Similar to the collaborative recommendation algorithm proposed for the PLSA model, we also incorporate the usage knowledge derived by LDA model into a collaborative filtering algorithm. We first partition user sessions into various session clusters based on the calculated posterior probability values, in turn, view the centroids of the session clusters as the representatives of the usage patterns. Integrating the usage knowledge into the proposed weighted scoring scheme eventually generates the recommended page list.

Case Studies of Gait Pattern Mining In this research, we aim to extend the developed methodologies and technologies to a biomechanical data mining application, i.e. gait pattern mining. Gait analysis is an important topic in the movement clinical research and application for different specific populations, such as CP patients or elderly people. In this study, we conduct the following case studies:

- Case study of CP gait pattern mining using the traditional clustering based algorithms. We develop standard k-means and hierarchical clustering based approaches to find CP-specific gait patterns. The gait characteristics of healthy children and CP patients at different pathological level are modelled by different gait vectors of kinematic variables, i.e. temporal-distance parameters. The discovered gait pattern knowledge can provide a useful means for researchers or clinicians to monitor the development of CP or assess the effectiveness of the intervention.

– Case study for monitoring the fall risk of elderly population using a SOM-based clustering approach. We employ a SOM-based clustering algorithm to investigate the locality of the gait in a transformed SOM grid map. The derived SOM grid could offer us a visualized representation of gait patterns for screening the fall risk in an elderly population.

1.4. Outline of the Dissertation

This dissertation contains nine chapters. Figure 1-2 illustrates the structure of the thesis chapters, where TO denotes Task-Oriented and UP stands for User Profiling. Form the figure, it is seen, after introducing the research problems and some mathematical concepts, formulas and algorithms used in the later chapters, we aim to employ three kinds of latent semantic analysis models to address Web usage mining in the following three chapters; in chapter 6 and 7, we investigate using the discovered usage knowledge for Web recommendations. Meanwhile, two application investigations of gait pattern mining are carried out in chapter to evaluate the applicability of the developed methodologies and technologies in chapter 8. We eventually summarize the claims of the thesis and indicate some future research directions in chapter. In a summary, we put forward a mathematical foundation first, then conduct various specific knowledge discovery tasks followed a task of knowledge application, which are logically bunched into a whole data mining cycle. The remainder of the dissertation is organized as follows.

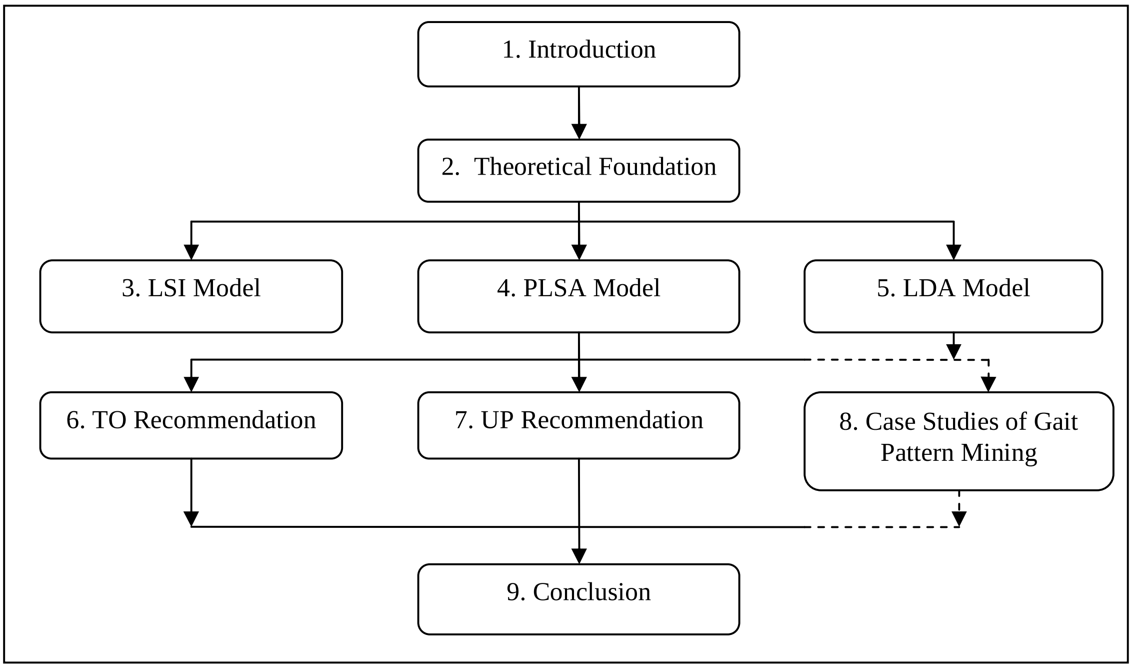


Figure 1-2. The structure of the thesis

Chapter 2 describes the basic concepts and techniques necessary for Web usage mining and Web recommendations. A mathematical usage data model (matrix) is presented in this chapter, and the related mathematical knowledge and background are provided for better understanding the algorithms and techniques developed in this dissertation. The developed algorithms and techniques for Web usage mining, latent semantic analysis and Web recommendation are also reviewed and discussed based on which this dissertation develops. This chapter provides a foundation for further study of Web usage mining and Web recommendation described in the following chapters.

In chapter 3, we address the Web usage mining by employing the traditional LSI approach. A LUI algorithm is proposed to extract the latent semantic knowledge from the usage data via computing the singular values of the original usage data, which approximates the original semantics hidden in the usage matrix. Apart from other algorithms, such as [29], that employed a standard clustering algorithm on the usage data directly to find the aggregates of user sessions, we develop another algorithm that performs clustering on the transformed usage space to improve the Web usage mining.

In this algorithm, each user session is represented by a dimensionality-reduced page vector, which conveys the latent semantic relationships among the Web objects in the usage data model. From the revealed relationships, user session aggregates that contain highly semantic similarity are eventually generated. Experiments are conducted to demonstrate the effectiveness of the algorithm for usage pattern mining.

Chapter 4 presents an alternative latent semantic analysis model, the PLSA model. In contrast to the tradition LSI approaches, the PLSA model is based on a more solid foundation of statistical analysis. It is capable of discovering the latent semantic factor space associated with the usage patterns in addition to the traditional latent semantic analysis. In this chapter, a series of equations are formulated based on Bayesian and uncertainty theory, which characterize the associations between Web objects (i.e. Web pages and user sessions) and latent semantic factors. Meanwhile, an EM algorithm is developed to estimate the parameters of the PLSA model that leads to a maximum likelihood of the usage data. The parameters of the PLSA model are termed as a set of conditional probability distribution of Web pages or user sessions against the latent semantic factors, which convey the intrinsic aggregation property of the Web objects. We then utilize these factor-based feature vectors to group Web pages and user sessions as well as identify the latent semantic factor space via a probability inference approach. In particular, two sets of similarity measures of Web pages and user sessions are proposed. The effective of the algorithm is shown by a series of experiments.

In chapter 5, we address the Web usage mining by applying a novel generative model, i.e. LDA model, which is also an alternative latent semantic analysis model. We first systematically summarize the evolution of the generative models, and intensively discuss the strength of the analytical model employed. We then describe the algorithm of a variational EM algorithm to calculate the parameters of LDA model. The parameters in terms of posterior probability and Dirichlet value are used to derive user access patterns. We carry out an experimental analysis to evaluate the effectiveness and efficiency of the proposed analytical models.

We turn to address Web recommendations in chapter 6 by using the usage knowledge discovered based on the above analytical models. We first introduce a top-N weighted scoring scheme, which forms the common base of various Web recommendation algorithms. This algorithm is to compute the recommendation score of each page based on the probability weight, which represents the likelihood being visited. In this chapter, we also present a Web recommendation algorithm by identifying user task preference distributions and integrating the latent task space into the collaborative filtering approach. Analysis of the very first clicks on Web pages results in capturing the task-driven probability distribution of one user session over task space, in turn, determines the predominant tasks having significant probability values. We eventually calculate the recommendation scores by integrating the predominant tasks with the discovered task representatives. The evaluation is done by a series of experiments on real world log files.

Chapter 7 concentrates on the study of employing a user profiling approach for Web recommendations. In this chapter, we utilize the proposed user profile approach to deal with Web recommendations based on the PLSA and LDA models. The user profiles are applied into the collaborative recommendation algorithm to select the most matched usage pattern and predict the most potentially visited pages by referring to the visiting histories of other users who exhibit similar navigation preferences. Experimental results on two Web log files show the effectiveness of the proposed algorithms.

We extend the developed technologies and methodologies to two case studies of gait pattern mining in chapter 8. First we address discovering gait patterns of CP patients, which are represented by the attribute vectors of the temporal-distance kinematic gait variables. The CP gait pattern mining algorithm is implemented by employing the traditional clustering algorithms, i.e. k-means and hierarchical clustering algorithms. Gait patterns are derived by the centroids of the gait clusters, in turn, treated as the diagnostic indicatives for assessing the walking impairment of the CP patients. In the second part of this chapter, we develop a SOM-based clustering algorithm to address gait analysis for monitoring fall risk of elderly population. By using a specific gait variable, *Minimum Foot Clearance* (MFC) to model elder people walking characteristics, we construct a gait data model in terms of various statistical parameters of the MFC variable. Then we employ a SOM-based clustering algorithm on a gait dataset which consists of three groups of gait data, i.e. younger subjects, elderly but healthy subjects and elderly subjects with impaired walking ability, to separate these subjects into different gait groups. In the transformed gait SOM grid, it is shown there are various groups of subjects assigned to different portions of the figure. The locality of the aggregation indicates the gait pattern knowledge. Meanwhile, the centroids of the clusters stand for the characteristics of the gait information. Experimental results are visualized and tabulated to show the application potential of gait pattern mining with the proposed approaches.

We conclude the dissertation and outline possible future work in chapter 9.

2. Fundamentals of Web Data Mining and Web Recommendation

2.1. Introduction

It is well known that Internet has become a very popular a powerful platform to store, disseminate and retrieve information as well as a data respiratory for knowledge discovery. However, Web users always suffer from the problems of information overload and drowning due to the significant and rapid growth in the amount of information and the number of users. The problems of low precision and low recall rate caused by above reasons are two major concerns that users have to deal with while searching for the needed information over the Internet. On the other hand, the huge amount of data/information residing over the Internet contains a large amount of valuable informative knowledge that could be discovered via advanced data mining approaches. It is believed that mining this kind of knowledge will greatly benefit Web site designs and Web application developments, and promote other related applications, such as business intelligence, e-Commerce, and entertainment broadcast etc. Thus, the emerging of Web has put forward a large number of challenges to Web researchers for web-based information management and retrieval. Web researchers and engineers are requested to develop more efficient and effective techniques to satisfy the demands of Web users.

Web data mining is one kind of these techniques that efficiently handle the tasks of searching the needed information from the Internet, improving the Web site structure to provide better Internet service quality and discovering the informative knowledge from the Internet for advanced Web applications. In principle, Web mining techniques are the means of utilizing data mining methods to induce and extract useful information from Web data and services. Web mining research has attracted a variety of academics and researchers from database management, information retrieval, artificial intelligence research areas especially from knowledge discovery and machine learning, and many research communities have addressed this topic in recent years due to the tremendous growth of data contents available on the Internet and the urgent needs of e-commerce applications especially. Dependent on various mining targets, Web data mining could be categorized into three types of Web content, Web structure and Web usage mining. In this study, we focus on Web usage mining: that is, discovering user access pattern knowledge from Web log files, which contain the historic visiting records of users on the website. The discovered usage knowledge makes it possible for Web designers and developers to better understand user navigational behaviour, which will not only provide them with helps in re-structuring Web sites, but also improve the Web presentations.

Web recommendation or personalization is a process that utilizes the informative knowledge learned from Web data mining as a knowledge base, then predicts user potential access preferences, and recommends the customized Web contents by referring to the knowledge base. The knowledge base can be made up of content, linkage, usage and semantic information. Recommender systems are well studied in the context of artificial intelligence and information retrieval. To-date, there are two kinds of approaches and techniques commonly used in recommender systems, namely content-based filtering and collaborative filtering systems [38, 39]. Recently collaborative filtering approaches have been extensively used in Web recommendation applications and have achieved great success as well [42-44]. Meanwhile, with the progress of the

Web usage mining research, Web researchers intend to combine the usage pattern knowledge into the recommendation to improve Web recommendation systems. Using the usage knowledge as a collaborative information source will dramatically improve the recommendation performance and the online response efficiency. With the benefit of great progress in data mining research communities, many data mining techniques, such as collaborative filtering based on the k-Nearest Neighbour algorithm (*kNN*) [42-44], Web user or page clustering [29, 45, 46], association rule mining [47, 48] and sequential pattern mining technique [19] have been adopted in current Web usage mining methods.

To implement Web usage mining efficiently, it is essential to first introduce a solid mathematical framework, on which the data mining/analysis is performed. There are many types of data expressions could be used to model the co-occurrence of Web user behaviours, such as matrices, directed graphs and click sequences and so on. Different data expression models have different mathematical and theoretical backgrounds. In particular, we aim to adopt the commonly used matrix expression, which is also widely used in Web structure (linkage) mining, in this study. In this framework, the user navigational behaviour is modelled by a usage matrix, in the form of the session-page vector. Based on the proposed mathematical framework, a variety of data mining and analysis operations can be employed to conduct Web usage data mining. Clustering is a main analytical approach used in this work, which is to partition Web objects into various groups based on their mutual distance. On the other hand, the latent semantic analysis model is another focus of Web usage mining, which is able to discover the underlying relationships among the Web objects. Some basic descriptions regarding their algorithms and concepts are discussed in this chapter.

In the context of Web recommendation, there are several algorithms and techniques, which have been studied and developed in conventional recommender systems. In this chapter, we also review and discuss the background involved in Web recommendations. All these fundamentals prepare us a necessary knowledge base for better understanding the studies addressed.

The remainder of this chapter is organized as follows: we first introduce a Web usage data model in the form of the matrix expression in section 2.2. Clustering is discussed in section 2.3. We review the theoretical background of the latent semantic analysis in section 2.4, and algorithms and similarity measures with respect to Web recommendations are given in section 2.5.

2.2. Web Data Model and Matrix Expression

For efficient Web data management, the Web data model is essential and crucial, on which a variety of data mining and machine learning techniques are employed. To achieve the desired mining tasks discussed above, there are different Web data models in the forms of feature vectors engaged in pattern discovery and knowledge application. According to the three identified categories of Web mining methods, three types of Web data/sources, namely content data, structure data and usage data, are mostly considered in the context of Web mining. Before we start to propose different Web data models, we firstly give a brief discussion on these three data types in the following paragraphs.

Web content data is a collection of objects used to convey content information of Web pages to users. In most cases, it is composed of textural material and other types of multimedia contents, which include static HTML/XML pages, images, sound and video files, and dynamic pages generated from scripts and databases. The content data also includes semantic or structured meta-data embedded within the site or individual pages. In addition, the domain ontology might be considered as a complementary type of content data hidden in the site implicitly or explicitly. The underlying domain knowledge could be incorporated into Web site designs in an implicit manner, or be represented in some explicit forms. The explicit form of domain ontology can be conceptual hierarchy e.g. product category, and structural hierarchy such as yahoo directory etc [71].

Web structure data is a representation of linking relationships between Web pages, which reflects the organization concept of a site from the viewing point of the designer [34]. It is normally captured by the inter-page linkage structure within the site, thus, is called linkage data. Particularly, the structure data of a site is usually represented by a specific Web component, called "site map", which is generated automatically when the site is completed. For dynamically generated pages, the site mapping is becoming more complicated to perform since more techniques are required to deal with the dynamic environment.

Web usage data is mainly sourced from Web log files, which include Web server access logs and application server logs [28, 72]. The log data collected at Web access or application servers reflects navigational behaviour knowledge of users in terms of access patterns. In the context of Web usage mining, usage data that we need to deal with is transformed and abstracted at different levels of aggregations, namely Web page sets and user session collections. Web page is a basic unit of a Web site organization, which contains a number of meaningful units serving for the main functionality of the page. Physically, a page is a collection of Web items, generated statically or dynamically, contributing to the display of the results in response to a user action. A page set is a collection of whole pages within a site. User session is a sequence of Web pages clicked by a single user during a specific period. A user session is usually dominated by one specific navigational task, which is exhibited through a set of visited relevant pages that contribute greatly to the task conceptually. The navigational interest/preference on one particular page is represented by its significant weight value, which is dependent on user visiting duration or click number. The user sessions (or called usage data), which are mainly collected in the server logs, can be transformed into a processed data format for the purpose of analysis via a data preparing and cleaning process. In one word, usage data is a collection of user sessions, which is in the form of weight distribution over the page space.

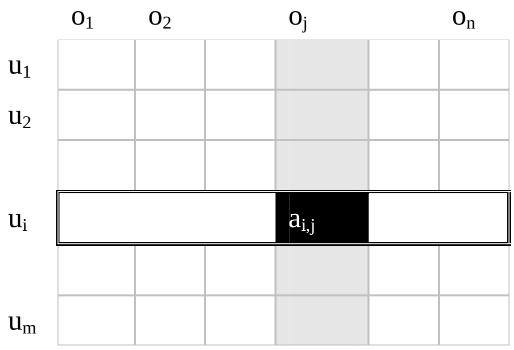


Figure 2-1. The illustration of Web data model

Matrix expression has been widely used to model co-occurrence activities like Web data. The illustration of a matrix expression for Web data is shown in Figure 2-1. In this scheme, the rows and columns correspond to various Web objects which are dependent on various Web data mining tasks. In the context of Web content mining, the relationships between a set of documents and a set of keyword could be represented by a document-keyword co-occurrence matrix, where the rows of the matrix represent the documents, while the columns of the matrix correspond to the keywords. The intersection value of the matrix indicates an occurrence rate of a specific keyword appeared in a particular document, i.e. if a keyword is appeared in a document, the corresponding matrix element value is 1, otherwise 0. Of course, the element value could also be a precise weight rather than 1 or 0 only, which exactly reflects the occurrence degree of two concerned objects of document and keyword. For example, the element value could represent a frequent rate of a specific keyword in a specific document. Likewise, to model the linkage information of a Web site, an adjacency matrix is used to represent the relationships between pages via their hyperlinks. And usually the element of the adjacency matrix is defined by the hyperlink linking two pages, that is, if there is a hyperlink from page i to page j (i ≠ j), then the value of the element a_{ij} is 1, otherwise 0. Since the linking relationship is directional, i.e. given a hyperlink directed from page *i* to page *j*, the link is an out-link for *i*, while an in-link for *j*, and vice versa. In this case, the i^{th} row of the adjacency matrix, which is a page vector, represents the out-link relationships from page *i* to other pages; the j^{th} column of the matrix represents the in-link relationships linked to page *i* from other pages.

In Web usage mining, we can model one user session as a page vector in a similar way. The user access interest exhibited may be reflected by the varying degree of visits on different Web pages during one session. Thus, we can represent a user session as a collection of pages visited in the period along with their significant weights. The total collection of user sessions can, then, be expressed a usage matrix, where the i^{th} row is the sequence of pages visited by user *i* during this period; and the j^{th} column of the matrix represents the fact which users have clicked this page *j* in server log files. The element value of the matrix, a_{ij}, reflects the access interest exhibited by the user *i* on the page *j*, which could be used to derive underlying access patterns of users.

The mathematical framework of Web data mining and Web recommendation is depicted in Figure 2-2, which outlines the schematic structure of the proposed prototype. It is shown that Web data model, Web mining and recommendation algorithm are the three main components of the whole scheme.

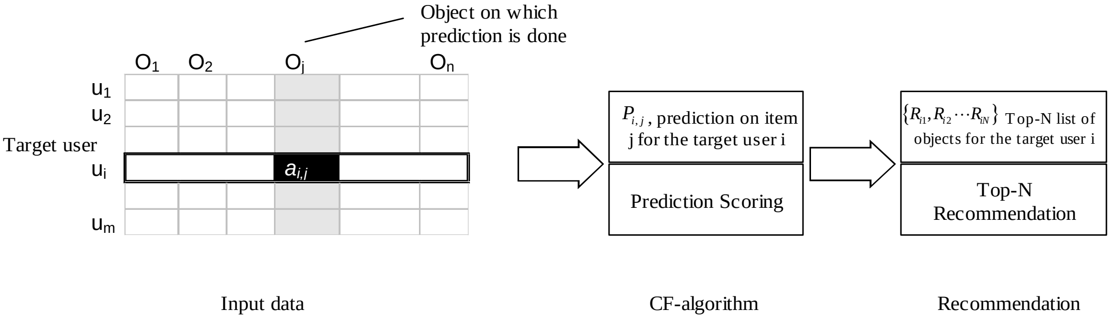

Object on which prediction is done
O_1
O_2
O_j
O_n
u_1
u_2
Target user
u_i
u_m
$a_{i,j}$
$P_{i,j}$, prediction on item j for the target user i
Prediction Scoring
$\{R_{i1}, R_{i2} \cdots R_{iN}\}$ Top-N list of objects for the target user i
Top-N Recommendation
Input data
CF-algorithm
Recommendation

Figure 2-2. A framework for Web usage mining and Web recommendation

In this study, we mainly focus on introduction of Web usage data model. Given *n* Web pages in a Web site and *m* Web user sessions recorded in a Web log file for a specific period, after data pre-processing, which consists of page identification and user sessionization processes [73], we can built up a set of *n* pages as $P = \{p_1, p_2, \ldots p_n\}$ and a set of *m* user sessions as $S = \{s_1, s_2, \ldots, s_m\}$. This process is executed in a similar way happened in information retrieval, in which a word vocabulary collection and document corpus are created in the form of a document-keyword matrix and each entry in this matrix is determined by the tf/idf value [61]. In the context of Web usage mining, therefore, the user session and Web page collection could be considered as the equivalent concepts of document corpus and keyword vocabulary sets in information retrieval. That is, each user session can be, in turn, expressed as a sequence of weight-page pairs,

$s_i = \{(p_1, a_{i1}), (p_2, a_{i2}), \dots (p_n, a_{in})\}$, where a_{ij} stands for a significant weight on the page p_j contributed by the user session s_i. By simplifying the above expression in terms of page vectors, each user session can be considered as an n-dimensional vector of pages, $s_i = \{a_{i1}, a_{i2}, \dots a_{in}\}$, where a_{ij} denotes the weight for the page p_j in the s_i user session. Figure 2-3 illustrates the constructed Web usage data model in a matrix expression, in which each row indicates a user session over the page set, while each column is corresponding to a weight distribution of one specific page over the session set.

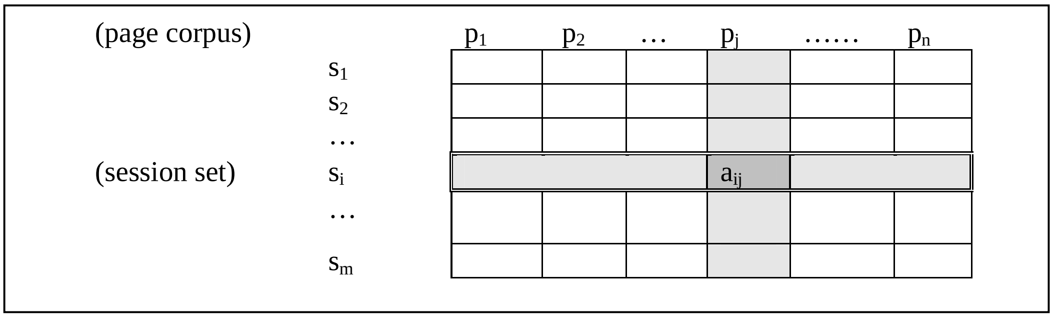


Figure 2-3. Web usage data model in a matrix expression

1) Main Movies: 20sec Movies News: 15sec NewsBox: 43sec Box-Office Evita: 52sec News Argentina:31 sec Evita: 44sec
2) Music Box: llsec Box-Office Crucible: 12sec Crucible Book: 13sec Books: 19sec
3) Main Movies: 33sec Movies Box: 21sec Boxoffice Evita: 44sec News Box: 53sec Box-office Evita: 61 sec Evita : 31sec
4) Main Movies: 19sec Movies News: 21sec News box: 38sec Box-Office Evita:61 sec News Evita:24sec Evita News: 31 sec News Argentina: 19sec Evita: 39sec
5) Movies Box: 32sec Box-Office News: 17sec News Jordan: 64sec Box-Office Evita: 19sec Evita: 50sec
6) Main Box: 17sec Box-Office Evita: 33sec News Box: 41 sec Box-Office Evita: 54sec Evita News: 56sec News: 47sec

$$SP_{ex} = \begin{bmatrix} 9.76 & 7.32 & 36.1 & 25.4 & 21.5 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 21.8 & 0.00 & 20.0 & 23.6 & 34.6 \\ 13.6 & 8.64 & 21.8 & 43.2 & 12.8 & 0.00 & 0.00 & 0.00 \\ 7.54 & 8.33 & 32.1 & 34.2 & 27.8 & 0.00 & 0.00 & 0.00 \\ 0.00 & 17.6 & 35.2 & 19.8 & 27.5 & 0.00 & 0.00 & 0.00 \\ 6.85 & 0.00 & 35.5 & 35.1 & 22.6 & 0.00 & 0.00 & 0.00 \end{bmatrix}$$

Figure 2-4: A usage snapshot and its session-page matrix expression

As a result, the whole user session data can be utilized to form Web usage data represented by a session-page matrix $SP_{m\times n} = \{a_{ij}\}$. The element value in the session-page matrix, a_{ij}, can be represented by a weight associated with the page p_j in the user session s_i, which is usually determined by the number of hits or the amount time spent on the specific page. Generally, the weight a_{ij} associated with the page p_j in the session s_i should be normalized across pages in the same user session in order to eliminate the influence caused by the relative amount difference of visiting time durations or hit numbers. The so-called session normalization implementation is capable of capturing the relative significance of a page within one user session with respect to other pages accessed by the same user. Figure 2-4 illustrates an example of a usage data snapshot, in which the names (in *italic*) are the titles of Web pages followed by the links and corresponding link times (underlined), and its corresponding session-page matrix in terms of normalized weight forms from [36, 56] is displayed as well.

2.3. Clustering Algorithms

Clustering analysis is a widely used data mining algorithm for many data management applications. Clustering is a process of partitioning a set of data objects into a number of object clusters, where each data object shares the high similarity with the other objects within the same cluster but is quite dissimilar to objects in other clusters. Different from classification algorithm that assigns a set of data objects with various labels previously defined via a supervised learning process, clustering analysis is to partition data objects objectively based on measuring the mutual similarity between data objects, i.e. via a unsupervised learning process. Due to the fact that the class labels are often not known before data analysis, for example, in case of being hard to assign class labels in large databases, clustering analysis is sometimes an efficient approach for analysing such kind of data. To perform clustering analysis, similarity measures are often utilized to assess the distance between a pair of data objects based on the feature vectors describing the objects, in turn, to help assigning them into different object classes/clusters. There are a variety of distance functions used in different scenarios, which are really dependent on the application background. For example, cosine function and Euclidean distance function are two commonly used distance functions in information retrieval and pattern recognition [61]. On the other hand, assignment strategy is another important point involved in partitioning the data objects. Therefore, distance function and assignment algorithm are two core research focuses that attract a lot of efforts contributed by various research domain experts, such as from database, data mining, statistics, business intelligence and machine learning etc.

The main data type typically used in clustering analysis is the matrix expression of data. Suppose that a data object is represented by a sequence of attributes/features with corresponding weights, for example, in the context of Web usage mining, a usage data piece (i.e. user session) is modelled as a weighted page sequence. Like what we discussed above, this data structure is in the form of the object-by-attribute structure, or an n-by-m matrix where *n* denotes the number of data objects and *m* represents the number of attributes. In addition to data matrix, similarity matrix, where the element value reflects the similarity between two objects is also used for clustering analysis. In this case, the similarity matrix is expressed by an n-by-n table. For example, an adjacency matrix addressed in Web linkage analysis is actually a similarity/relevance matrix. In this work, we adopt the first data expression, i.e. data matrix to address Web usage mining and Web recommendation.

To date, there are a large number of approaches and algorithms developed for clustering analysis in the literature [2, 17, 29, 35, 45, 56, 69, 74-76]. Based on the operation targets and procedures, the major clustering methods can be categorized as: Partitioning methods, hierarchical methods, density-based methods, grid-based methods, model-based methods, high-dimensional clustering and constraint-based clustering [17]. Partitioning method is to assign *n* objects into *k* predefined groups, where each group represents a data segment sharing the highest average similarity in comparison to other groups. The well-known *k*-means is one of the most conventional partitioning clustering algorithms. The algorithm is expressed as follows:

Step 1: arbitrarily choose *k* data points as initial cluster mean centres;

Step 2: then assign each data to the cluster with the nearest centres, and update each mean centre of cluster;

Step 3: repeat step 2 until all centres don’t change and no reassignment is needed;

Step 4: finally output subject clusters and their corresponding centres.

Hierarchical clustering is to construct a hierarchical tree of the given set of data objects instead of crisply classifying each data objects into a distinct data segment. The algorithm is executed in the following way:

Step 1: calculate the mutual distance of two data points (distance matrix) as the clustering criteria;

Step 2: decompose the dataset into a set of levels of the nested aggregations based on the distance matrix (i.e. the tree of clusters);

Step 3: cut the hierarchical tree at the desired level by selecting a predefined threshold, and then explicitly merge all connected subjects below the cut level to create various clusters;

Step 4: output the dendrograms and the clusters.

Hierarchical clustering provides an easily visualized way to modelling the underlying relationships among the data objects. Hierarchical clustering can be considered an agglomerative approach, which suffers from the problem of one-way construction, that is, it can not be undone during the hierarchical tree construction procedure.

Model-based methods hypothesize that there exists a model for each of the clusters, into which one data object is best fitted by measuring a density function. The density function that associates with the special distribution of the data helps to determine the cluster number and to assign the data objects into various clusters. For example, *Self Organizing Map* (SOM) based clustering is one of the model-based methods, which is to map a data object in a high-dimensional space into a low-dimensional (e.g. 2-D or 3-D) grid map via a neural network based algorithm. The SOM-based clustering algorithm is eventually to map the original data objects onto different data blocks/segments of the SOM grid and the locality of the data points indicates the visualized clustering information. The detailed description of SOM-based clustering algorithm is briefly summarized as follows:

Step 1: The SOM process consists of a regular, usually two-dimensional, grid of map units. Each unit *i* is represented by an n-dimensional prototype vector, $m_i = [m_{i1}, \cdots, m_{in}]$, where *n* is the dimension of the input space. In the grid, the units are connected to adjacent ones by a neighbourhood relation. The number of the map units, which varies depending on the size of the input space, determines the accuracy and generalization capability of the SOM;

Step 2: On each learning step, a data sample *x* is selected and the nearest map unit (i.e. *Best Matching Unit* - BMU) is found on the map. The prototype vector of the BMU and its neighbouring units on the grid are merged toward the sample vector:

$$m_i(t+1) = m_i(t) + \alpha(t)h_{bi}(t)[x - m_i(t)] \tag{2.1}$$

where *α(t)* is the learning rate and $h_{bi}(t)$ is a neighbourhood kernel centred on the winner unit. Both of learning units and neighbourhood kernels radius decrease monotonically with time;

Step 3: The SOM is trained iteratively until the following error function reaches the minimum:

$$E = \sum_{i=1}^{N}\sum_{j=1}^{M} h_{bj} \left\| x_i - m_j \right\|^2 \tag{2.2}$$

where *N* is the number of the training data and *M* is the number of the map units.

2.4. Latent Semantic Analysis Models

Different from clustering algorithm, latent semantic analysis is one kind of statistical data analytical models, which is to perform analysis on a so-called latent semantic space rather than on the original data matrix. The latent semantic space is usually a transformed data space derived from the original input space, which can convey the semantic information in some extents. *Latent Semantic Indexing* (LSI), one kind of LSA model, was firstly proposed to address finding semantic relevance in the context of information retrieval and digital library. Researchers utilized it to identify the semantic themes hidden in a large amount of document collections. LSI algorithm has achieved great success in text mining and has been extended to other related applications [25]. The standard LSI algorithm is based on a SVD operation. The SVD definition of a matrix is illustrated as follows [77]: For a real matrix $A = \left[a_{ij}\right]_{m\times n}$, without loss of generality, suppose $m \geq n$ and there exists a SVD of A (shown in Figure 2-5):

$$A = U\begin{pmatrix}\Sigma_1 \\ 0\end{pmatrix}V^T = U_{m\times m}\sum{}_{m\times n}V^T_{n\times n} \tag{2.3}$$

where U and V are orthogonal matrices $U^TU = I_m, V^TV = I_n$. Matrices U and V can be respectively denoted as $U_{m\times m} = [u_1, u_2, \cdots u_m]_{m\times m}$ and $V_{n\times n} = [v_1, v_2, \cdots v_n]_{n\times n}$, where $\mathrm{u_i}, (i = 1, \cdots, m)$ 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 vector $v_j = (v_{1j}, v_{2j}, \cdots v_{nj})^T$. Suppose $rank(A) = r$ and the singular values of A are the diagonal elements of $\sum$ as follows:

$$\sum = \begin{bmatrix}\sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_n\end{bmatrix} = diag(\sigma_1, \sigma_2, \cdots \sigma_m),$$

where $\sigma_i \geq \sigma_{i+1} > 0$, for $1 \leq i \leq r-1$; $\sigma_j = 0$, for $j \geq r+1$, that is

$\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_n = 0$

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

$A_k = U_k \sum{}_k V^T_k$

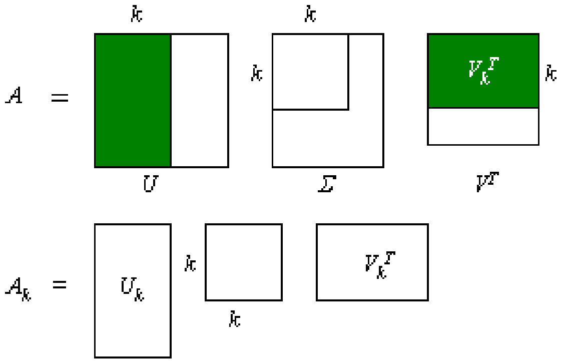


Figure 2-5. An Illustration of SVD approximation

As known from the theorem in algebra [77], A_k is the best approximation matrix to *A* and conveys the maximum latent information among the processed data. This property makes it possible to find out the underlying semantic association from the original feature space with a reduced computational cost, in turn, is able to be used for latent semantic analysis.

While SVD is usually used in the conventional LSI techniques, some variants of the LSA methods have been proposed recently in the context of Web information processing and text mining. Apart from the difference at the theoretical formulation, the common characteristics of these methods are to map the original feature space into a new transformed feature space, and maintain the maximum approximation of the original feature space. For example, PLSA and LDA model are two representatives of such kinds of approaches [67, 78]. More details regarding these two models will be intensively presented in the following chapter 5 and 6.

2.5. Recommendation Algorithms

As discussed in the introduction part, the aim of Web recommendation is to find the most matched user access pattern to the active user session, which is derived from Web usage mining, and to recommend a list of pages that the users might be interested in, via referring to the visiting preferences of the chosen usage pattern. To perform recommendations efficiently and effectively, there are a variety of machine learning algorithms that have been well studied and developed, and can be used in Web recommendation. In this section, we simply review several related algorithms that are often used in recommendation processes.

2.5.1. k-Nearest Neighbour Algorithm

K-Nearest-Neighbour (kNN) approach, which is to compare the current user activity with the historic records of other users for finding the top *k* users who share the most similar behaviours to the current one, is the most often used recommendation scoring algorithm in recommender systems. In conventional recommender systems, finding *k* nearest neighbours is usually accomplished by measuring the similarity in rating of items or visiting on Web pages between the current user and others. The found neighbouring users are then used to produce a prediction list of items that are potentially rated or visited but not done yet by the current active user via collaborative filtering approaches. Therefore, the core component of the kNN algorithm is the similarity function that is used to measure the similarity or correlation between users in terms of attribute vectors, in which each user activity is characterized as a sequence of attributes associated with corresponding weights.

A variety of similarity functions can be used as measuring metrics. Among these measures, Pearson correlation coefficient and cosine similarity are two well-known and widely used similarity functions in recommender systems [79].

Correlation-based Similarity

Pearson correlation coefficient, which is to calculate the deviations of users' ratings on various items from their mean ratings on the rated items, is a commonly used similarity function in traditional collaborative filtering approaches, where the attribute weight is expressed by a feature vector of numeric ratings on various items, e.g. the rating can be from 1 to 5 where 1 stands for the lest like voting and 5 for the most preferable one. The Pearson correlation coefficient can well deal with collaborative filtering since all ratings are on a discrete scale rather than on an analogous scale. The measure is described below. Given two users *i* and *j*, and their rating vectors R_i and R_j , the Pearson correlation coefficient is then defined by:

$$sim(i,j) = corr(R_i, R_j) = \frac{\sum_{k=1}^{n}\left(R_{i,k} - \overline{R_i}\right) \bullet \left(R_{j,k} - \overline{R_j}\right)}{\sqrt{\sum_{k=1}^{n}\left(R_{i,k} - \overline{R_i}\right)^2 \sum_{k=1}^{n}\left(R_{j,k} - \overline{R_j}\right)^2}} \tag{2.4}$$

where $R_{i,k}$ denotes the rating of the user *i* on the item *k*, $\overline{R_i}$ is the average rating of the user *i*.

However, this measure is not appropriate for the Web mining scenario where the data type encountered (i.e. user session) is actually a sequence of analogous page weights. To address this intrinsic property of usage data, the cosine coefficient is a better choice instead, which is to measure the cosine function of the angle between two feature vectors. Cosine function is widely used in information retrieval.

Cosine-Based Similarity

The cosine coefficient can be calculated by the ratio of the dot product of two vectors with respect to their vector norms. Given two vectors *A* and *B*, the cosine similarity is defined as:

$$sim(A,B)=\cos\left(\vec{A},\vec{B}\right)=\frac{\vec{A}\bullet\vec{B}}{\left|\vec{A}\right|\times\left|\vec{B}\right|} \tag{2.5}$$

where "·" denotes the dot operation and "×" denotes the norm form.

2.5.2. Content-Based Recommendation

Content-based recommendation is a textual information filtering approach based on user's historic ratings on items. In a content-based recommendation, a user is associated with the attributes of the items that rated, and a user profile is learned from the attributes of the items to model the interest of the user. The recommendation score is computed by measuring the similarity of the attributes the user rated with those not being rated, to determine which attributes might be potentially rated by the same user. As a result of attribute similarity comparison, this method is actually a conventional information processing approach in the case of recommendation. The learned user profile reflects the long-time preference of the user within a period, and could be updated as more different rated attributes representing the user's interest are observed. Content-based recommendation is helpful for predicting individual's preference since it is on the basis of referring to the individual's historic rating data rather than taking other's preferences into consideration.

2.5.3. Collaborative Filtering Recommendation

Collaborative filtering recommendation is probably the most commonly and widely used technique that has been well developed for recommender systems. As the name indicated, collaborative filtering recommendation in recommender systems works in a collaborative referring way that is to aggregate ratings or preferences on items, discover user profiles/patterns via learning from users' historic rating records, and generate a new recommendation on a basis of inter-pattern comparison. A typical user profile in recommender systems is expressed as a vector of the ratings on different items. The rating values could be either binary (like/dislike) or analogous-valued indicating the degree of preference, which is dependent on application scenarios. In the context of collaborative filtering recommendation, there are two major kinds of algorithms mentioned in literatures, namely memory-based and model-based collaborative filtering algorithms [39, 76, 79].

Memory-Based Collaborative Recommendation

Memory-based algorithms use the total ratings of users in training databases while computing recommendations. These systems can also be classified into two sub-categories: user-based and item-based algorithms [79]. For example, the user-based kNN algorithm, which is based on calculating the similarity between two users, is to discover a set of users who have a similar rating taste to the target user, i.e. neighbouring users, using the kNN algorithm. After k nearest neighbouring users are found, this system uses a collaborative filtering algorithm to produce a prediction of the top-N recommendations that the user may be interested in later. Given a target user u, the prediction on the item i is then calculated by:

$$p_{u,i} = \frac{\sum_{j=1}^{k}\left(R_{j,i} \bullet sim(u, j)\right)}{\sum_{j=1}^{k} sim(u, j)} \tag{2.6}$$

Here $R_{j,i}$ denotes the rating on the item *i* voted by the user *j*, and the only *k* most similar users (i.e. the k nearest neighbours of the user *i*) are considered in making recommendations.

In contrast to the user-based kNN, the item-based kNN algorithm [79, 80] is a different collaborative filtering algorithm, which is based on computing the similarity between two columns, i.e. two items. In an item-based kNN system, a mutual item-item similarity relation table is constructed first on a basis of comparing the item vectors, in which each item is modelled as a set of ratings by all users. To produce the prediction on an item *i* for the user *u*, it computes the ratio of the sum of the ratings given by the user on all items that are similar to the item *i* with respect to the sum of the involved item similarities as follows:

$$p_{u,i} = \frac{\sum_{j=1}^{k}\left(R_{u,j} \bullet sim(i, j)\right)}{\sum_{j=1}^{k} sim(i, j)} \tag{2.7}$$

Here $R_{u,j}$ denotes the prediction of the rating given by the user *u* on the item *j*, and only the *k* most similar items (k nearest neighbours of item *i*) are used to generate the prediction.

Model-based Recommendation

A model-based collaborative filtering algorithm is to derive a model from the historic rating data, and in turn, uses it for making recommendations. To derive the hidden model, a variety of statistical machine learning algorithms can be employed on the training database, such as Bayesian networks, neural networks, clustering and latent semantic analysis and so on. For example, in a model-based recommender system, a named *Profile Aggregations based on Clustering Transaction* (PACT) [29] clustering algorithm was employed to generate aggregations of user sessions, which are viewed as profiles via grouping users with a similar access taste into various clusters. The centroids of the user session clusters can be considered as access patterns/models learned from the Web usage data, in turn, used to make recommendations via referring to the Web objects visited by other users who share the most similar access task to the current target user.

Although existence of different recommendation algorithms in recommender systems, it is easily found that these algorithms are both executing in a collaborative manner, and the recommendation scores are dependent on the significant recommendation weights. In the following parts, we adopt these findings into our work.

3. Discovering Web Usage Pattern with Latent Semantic Indexing Approach

3.1. Introduction

Web clustering is one of the mostly used techniques in the context of Web mining, which is to aggregate similar Web objects, such as Web pages or users session, into a number of object groups via measuring their mutual vector distance. Basically, clustering can be performed upon these two types of Web objects, which results in clustering Web users or Web pages, respectively. The resulting Web user session groups are considered as representatives of user navigational behaviour patterns, while Web page clusters are used for generating task-oriented functionality aggregations of Web organizations. Moreover, the mined usage knowledge in terms of Web usage patterns and page aggregates can be utilized to improve Web site structure designs.

There has been a considerable amount of work on the applications of Web usage mining and recommender systems. For example, Mobasher et al [29] proposed an aggregate usage profile technique to cluster Web user transactions into various usage groups by using standard clustering algorithms, such as *k*-means clustering algorithm. On the other hand, an algorithm called PageGather [52] was proposed by Perkowith and Etzioni to discover significant page segments, which were used to help Web designers to add an additional index page that not existed before to facilitate Web users to locate their interested contents quickly, by using a Clique (complete link) clustering algorithm.

In the context of clustering, computational costs is a major concerned issue suffering researchers due to the particular characteristics of Web data, e.g. the problems of the high-dimension and the sparsity nature of Web data. For example, it is difficult, sometimes, to simply apply a standard clustering algorithm on the Web usage data with millions of user sessions to derive a collection of Web pages, which is resulting in a tough computational task. The reason is that instead of using pages as dimensions, the user sessions must be treated as dimensions and clustering is performed on this very high-dimensional space. To address there issues, dimensionality reduction techniques and alternative clustering algorithms are explored. Amongst these, *Latent Semantic Analysis* (LSA) is considered as an efficient dimensionality reduction algorithm with the latent semantic analysis capability, that is, the capability of discovering the hidden knowledge from Web data by taking the semantic property of data into consideration.

Latent Semantic Indexing (LSI), one kind of traditional LSA algorithms, is a statistical method, which is to reconstruct a co-occurrence observation space into a dimension-reduced latent space that keeps the maximum approximation of the original space by using mathematical transformation procedures such as *Singular Value Decomposition* (SVD). With the reduced dimensionality of the transformed data expression, the computational cost is significantly decreased accordingly, and the problem of sparsity of data is handled well as well. Besides, LSI based techniques are capable of capturing the semantic knowledge from the observation data, while the conventional statistical analysis approaches such as clustering or classification are in lack of finding underlying association among the observed co-occurrence. In last decades, LSI is extensively adopted in applications of information retrieval, image processing, Web research and data mining, and a large amount of successes have been achieved. In this chapter, we aim to integrate LSI analysis with Web clustering processes, to discover Web user session aggregates with better clustering quality, in other words, this techniques is on the basis of combination of latent semantic analysis and Web usage mining.

The remainder of this chapter is structured as follows. Section 3.2 first describes the theoretical background of LSI algorithm, which is based on a SVD calculation. Some basic concepts and formulas are presented to describe the details of the algorithm. We then propose an algorithm of Web clustering for building user profiles by incorporating latent semantic analysis in section 3.3. Experimental results are demonstrated in section 3.4. Related work and discussion are given in section 3.5. Finally we conclude this chapter in section 3.6.

3.2. Latent Semantic Indexing Algorithm

In this section, we first focus on introducing LSI algorithm and its related mathematical background, especially the knowledge of linear algebra in terms of Singular Value Decomposition operation, which forms the foundation of LSI algorithm. Upon the transformed semantic space, we propose a novel similarity function to measure the distance between two user sessions, which would be used in Web clustering.

3.1.1. Web Usage Data Model

The Web usage data is originally collected and stored in Web sever logs of websites, and is pre-processed for data analysis after performing data cleaning, page identification, and user identification for constructing the co-occurrence observation. In this study, we are mainly interested in the refined usage data instead of the raw data, more details regarding data preparation steps could be found in [73].

At this stage, we first review the usage data model described in the previous chapter, and particularly introduce the concept of the session-page matrix for Web usage mining.

As discussed above, in the context of Web usage mining, we construct two sets of Web objects: Web session set $S = \{s_1, s_2, \ldots s_m\}$ and Web page set $P = \{p_1, p_2, \ldots p_n\}$.

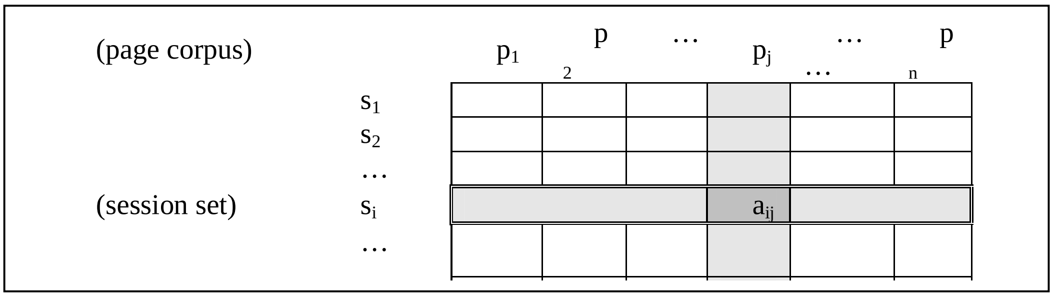


Figure 3-1. The schematic structure of a session-page matrix

And each user session is considered as a sequence of page-weight pairs, say $s_i = \{(p_1, a_{i1}), (p_2, a_{i2}), \ldots (p_n, a_{in})\}$. For the reason of simplicity expression, each user session can be re-written as a sequence of weights over the page space, i.e. $s_i = \{a_{i1}, a_{i2}, \ldots a_{in}\}$, where a_{ij} denotes the weight for the page p_j in the s_i user session. As a result, the whole user session data can be formed as a Web usage data matrix represented by a session-page matrix $SP_{m \times n} = \{a_{ij}\}$ (Figure 3-1 illustrates the schematic structure of the session-page matrix).

The entry value in the session-page matrix, a_{ij} is usually determined by the number of hits 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, a normalization manipulation across page space in the same user session is performed. Figure 3-2 illustrates two snapshots of Web log records extracted from a Web access log, in which each field is separated by a space. Particularly, note that the first and fourth fields are identified as the visitor IP address and the requested URL respectively, and are utilized to help collecting usage data. Thus, the first field can be identified as user session ID and the fourth attribute is treated as the page ID.

```
   202.161.108.167 - - [01/Feb/2003:00:00:03 +1100] "GET/timetables/city/2003s1/cc
4logo.gif HTTP/1.1" 206 14102 "http://www.cs.rmit.edu.au/timetables/city/2003s1/
cover.html "Mozilla/4.0 (compatible; MSIE 5.5; Windows 98)"

   213.183.13.65 - - [01/Feb/2003:00:00:16 +1100] "GET/˜winikoff/palm/dev.html HTT
P/1.1" 302 244 "http://www.google.de/search?q=sources+onboardc+examples&ie=UTF-
8&oe=UTF-8&hl=de&meta=" Scooter/3.3"
```

Figure 3-2. Snapshots from a Web access log

Once the usage matrix is constructed, we may applying conventional clustering algorithms on the user session data to classify user sessions into various groups, within which the classified sessions share the similar access interest. It is intuitive to perform clustering algorithms directly on each row vector of the usage matrix to determine the relative “close” session cluster by using a similarity-based measure, such as the commonly adopted cosine similarity from Information Retrieval. In [29], for example, an algorithm named PACT is proposed to address usage pattern mining based on the above mentioned technique. However, this kind of clustering technique only captures the mutual relationships between session data explicitly, it is incapable of revealing the “deeper” underlying characteristics of usage patterns. In this work, we propose an algorithm, named *Latent Usage Information* (LUI) to group user sessions semantically by taking the latent semantic information into account. For better understanding LUI algorithm, we first discuss some theoretical backgrounds of the SVD algorithm.

3.1.2. Singular Value Decomposition Algorithm

The SVD definition of a matrix is illustrated as follows [77]: For a real matrix $A = \left[a_{ij} \right]_{m \times n}$, without loss of generality, suppose $m \geq n$ and there exists a SVD of A:

$$A = U_{m \times m} \sum_{m \times n} V_{n \times n} \tag{3.1}$$

where U and V are orthogonal matrices. Matrices U and V can be denoted as $U_{m \times m} = [u_1, u_2, \ldots u_m]_{m \times m}$ and $V_{n \times n} = [V_1, V_2, \ldots, V_n]_{n \times n}$, where u_i (i = 1,…,m) is a m-dimensional vector $u_i = \{u_{1i}, u_{2i}, \ldots u_{mi}\}^T$ and v_j (j = 1,…,n) is a n-dimensional matrix $v_j = \{v_{1j}, v_{2j}, \ldots v_{nj}\}^T$. Suppose $rank(A) = r$ and singular values of A are the diagonal elements of $\sum$ as follows:

$$\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_n = 0 \tag{3.2}$$

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

Known from the theorem in algebra [77], A_k is the best approximation matrix to A and conveys the latent semantic information among the usage data. This property makes it possible to find out relative “close” user sessions at the semantic latent level based on their mutual similarity.

3.1.3. Representation of User Session in Latent Semantic Space

Once the SVD implementation is completed, we may rewrite user sessions with the obtained approximation matrix U_k, $\sum k$ and V_k by mapping them into another k-

dimensional latent semantic space. For a given session s_i, it is represented as a coordinate vector with respect to pages, written as $s_i = \{a_{i1}, a_{i2}, \ldots, a_{in}\}$. The projection of coordinate vector s_i in the *k*-dimensional latent semantic subspace is re-parameterized as

$$s_i^{'} = s_i V_k \sum{}_k = (t_{i1}, t_{i2}, ..., t_{ik}) \tag{3.3}$$

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

3.1.4. Similarity Measure

We adopt the traditional cosine function to capture the 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 a *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 transformed user sessions is defined as:

$$sim(s_i^{'}, s_j^{'}) = (s_i^{'} \cdot s_j^{'}) / \|s_i^{'}\|_2 \|s_j^{'}\|_2 \tag{3.4}$$

3.3. Latent Usage Information Algorithm

In this section, we present an algorithm called *latent Usage Information* (LUI) for clustering Web sessions and generating user profiles based on the discovered clusters. This algorithm consists of two steps, the first step is a clustering algorithm, which is to cluster the converted latent usage data into a number of session groups; and the next step is about generating a set of user profiles, which are derived from calculating the centroids of the discovered session clusters.

3.3.1. Algorithm for Clustering User Session

Here we adopt a *k*-means clustering algorithm, named MK-means clustering, to partition user sessions based on the transformed usage data matrix over the latent *k*-dimensional space. This algorithm does not need to predefine parameter *k* and *k* initial centroids, whereas the standard *k*-means has to do so to start clustering. The algorithm is described as follows:

[Algorithm 3.1]: *MK*-means clustering

[Input]: A converted usage matrix *SP* and a similarity threshold ε

[Output]: A set of user session clusters $SCL = \{SCL_i\}$ and corresponding centroids $Cid = \{Cid_i\}$

Step 1: Choose the first user session $s_i^{'}$ as the initial cluster SCL_1 and the centroid of this cluster, i.e. $SCL_1 = \{s_i^{'}\}$ and $Cid_1 = s_i^{'}$.

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

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

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

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

3.3.2. Building User Profile

As we mentioned above, each user session is represented as a weight-based page vector. In this way, it is reasonable to derive the centroid of the cluster obtained by the described clustering algorithm as a user profile. In this work, we compute the mean vector to represent the centroid. For each session cluster $SCL_i \in SCL$, the mean page vector of all sessions in the cluster (i.e. centroid), is determined by the ratio of the sum of page weights in SCL_i to the number of sessions in the cluster. In order to eliminate the impact of difference in visiting time or click numbers of each session, the weights are normalized while calculating the centroid of cluster. That is, the maximum weight in the constructed user profile is tuned to be 1, whereas other page weights are divided by the maximum weigh accordingly. Meanwhile, some less-contributed pages (i.e. those with mean weights being less than one certain limit) are filtered out. The algorithm for constructing user profile is as follows:

[Algorithm 3.2]: Building user profile based on LSI

[Input]: A set of user session clusters $SCL = \{SCL_k\}$

[Output]: A set of user profiles $UP = \{up_k\}$

Step 1: For each page *p* in the cluster SCL_k, we compute the mean weight value of pages

$$wt(p, SCL_k) = 1/|SCL_k| \sum_{s \in SCL_k} w(p,s) \tag{3.5}$$

where $w(p,s)$ is the weight of the page p in the session $s \in SCL_k$, and $|SCL_k|$ denotes the session number in the cluster SCL_k.

Step 2: For each cluster, furthermore, we calculate its mean vector (i.e. centroid) as

$$mv_C = \{< p, wt(p, SCL_k) > \mid p \in P\} \tag{3.6}$$

Step 3: For each page within the cluster, if the value is less than the threshold μ, the corresponding page will be filtered out, otherwise be kept.

Step 4: Sort pages with their weights in a descending order and output the mean vector as the user profile.

$$up_k = \{< p_{1k}, wt(p_{1k}, SCL_k) >, < p_{2k}, wt(p_{2k}, SCL_k) > ..., < p_{tk}, wt(p_{tk}, SCL_k)\} \tag{3.7}$$

where $wt(p_{1k}, SCL_k) > wt(p_{2k}, SCL_k) > \cdots > wt(p_{tk}, SCL_k) > \mu$.

Step 5: Repeat step 1 to 4 until all session clusters are processed, output user profiles.

3.4. Experimental Results

In order to evaluate the effectiveness of the proposed LUI algorithm, which consists of the Web clustering algorithm and the user profile generating algorithm, and evaluate the discovered user access patterns, we conduct experiments on two real world data sets and make comparisons with the previous work.

3.4.1. Experimental Design and Data Sets

We take two Web log files, which are public to access on the Internet for the purpose of research, as the usage data for experiments. These data sets are either in a raw data format or a pre-processed format. The first data set used in this study is downloaded from a KDDCUP website (www.ecn.purdue.edu/kddcup/). The data set is a commonly-used data source provided to test and compare knowledge discovery methods (prediction algorithm, clustering approaches, etc.) for the data mining purpose. Data pre-processing 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. Support filtering technique is used to eliminate these user sessions, leaving the only sessions with at least four pages. After data preparation, we have setup a data set including 9308 user sessions and 69 pages, where each session consists of 11.88 pages in average. We refer to this data set as “KDDCUP data”. In this data set, the entries in the session-page matrix associated with the specific page in a given session are determined by the numbers of Web page hits by a given user.

The second data set is from a university website log file and was made available by the author in [8]. The data is based on a random collection of users visiting this site for a 2-week period during April of 2002. After data pre-processing, 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 entry in the table corresponds to the amount of time (in seconds) spent on a page during a given session. For convenience, we refer to this data as “CTI data”. For each dataset, we randomly choose 1000 user sessions as the evaluation set, whereas the remainder part is selected as the training set for constructing user profiles.

The whole experiment design is structured as follows. We use the constructed usage data in the form of a matrix as a input data source, and apply appropriate data mining or analysis algorithms on it to extract usage knowledge and latent semantic relationships, which are formed as a usage knowledge base. Eventually we employ the developed algorithms along with the knowledge base to make Web recommendations. To assess the employed algorithms and data analytical models, we introduce some evaluation metrics and carry out comparisons with other related studies.

Other experiments performed in this thesis all follow this experimental design with various data analytical models. In this chapter, we aim to address usage pattern mining with LSI approach. The experimental results are presented in the following parts.

3.4.2. Results of User Profiles

We first utilize LUI algorithm to conduct Web usage mining on the selected two usage datasets respectively. We tabulate some results in below Table 3-1 and Table 3-2. In these tables, each user profile is represented by a sequence of significant pages together with corresponding weights. As we indicated before, the calculated weight is expressed in a normalized form, that is, the biggest value of them is set to be 1 while others are the relatively proportional values, which are always less than 1.

Table 3-1. Examples of generated user profiles from KDD dataset

Page #	Page 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-express_Checkout	0.38
65	Main-welcome	0.33
32	Main-registration	0.32
45	Checkout-confirm_order	0.26
Page #	Page 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 3-2. Examples of generated user profiles from CTI dataset

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

Table 3-1 depicts 2 user profiles generated from KDD dataset using LUI approach. Each user profile is listed in an ordered page sequence with corresponding weights, which means the greater weight a page contributes, the more likely it is to be visited. The first profile in Table 3-1 represents the activities involved in online-shopping behaviours, such as login, shopping cart, and checkout operation etc, especially occurred in purchasing leg-wear products, whereas the second user profile reflects the customers' concern with regard to the department store itself.

Analogously, some informative findings can be obtained in Table 3-2, which is derived from CTI dataset. In this table, three profiles are generated: the first one reflects the main topic of international students concerning issues regarding applying for admission, and the 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.

Looking at the generated user profile examples, it is shown that most of them do reflect one specific navigational intention, but some may represent more than one access themes.

3.4.3. Quality Evaluation of User Session Clusters

When the user session clustering is accomplished, we obtain a number of session clusters. However, how to assess the quality of the obtained clusters is another big concern for us in Web usage mining. A better clustering result should be that the sessions within the same cluster aggregate closely enough but keeping far from other clusters enough. After completing user session clustering, the next goal is to evaluate the quality of the generated clusters.

In order to evaluate the quality of clusters derived by LUI approach, we adopt one specific metric, named *Weighted Average Visit Percentage* (WAVP) [29]. This evaluation method is based on assessing each user profile individually according to the likelihood that a user session which contains any pages in the session cluster will include the rest pages in the cluster during the same session. The calculating procedure of WAVP metric is discussed as follows: suppose *T* is one of transaction set within the evaluation set, and for s specific cluster *C*, let *Tc* denote a subset of *T* whose elements contain at least one page from *C*. Moreover, the weighted average visit percentage of *Tc* may conceptually be determined by the similarity between *Tc* and the cluster *C* if we consider the *Tc* and *C* as in the form of page vector. Therefore, the WAVP is computed as:

$$WAVP = (\sum_{t \in T_c} \frac{\vec{t} \cdot \vec{C}}{|T_c|}) \Big/ (\sum_{p \in Pf} wt(p, pf)) \tag{3.8}$$

From the definition of WAVP, it is known that the higher the WAVP value is, the better the quality of obtained session cluster possesses.

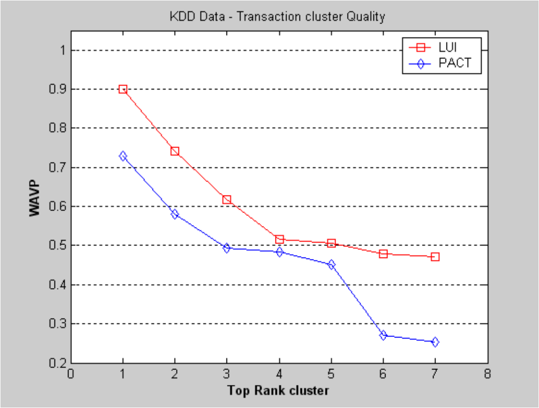


Figure 3-3. User cluster quality analysis results in terms of WAVP for KDD dataset

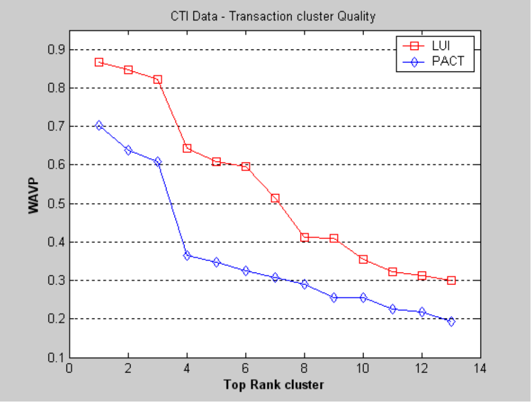


Figure 3-4. User cluster quality analysis results in terms of WAVP for CTI dataset

To compare the effectiveness and efficiency of the proposed algorithm with existing algorithms, here we use the PACT algorithm. We conduct data simulations upon two real world datasets by using these two approaches. Figure 3-3 and Figure 3-4 depict the comparison results in terms of WAVP values for KDD and CTI datasets with PACT respectively. In each figure, the obtained user profiles are arrayed in a descending rank according to their WAVP values, which reflect the quality of various clustering algorithms. From these two curves, it is easily concluded that the proposed LUI-based technique overweighs the standard *k*-means based algorithm in term of WAVP parameter. This is mainly due to the distinct latent analysis capability of LUI algorithm. In other words, LUI approach is capable of capturing the latent relationships among Web transactions and discovering user profiles representing the actual navigational patterns more effectively and accurately.

3.5. Related Work and Discussion

In the context of Web usage mining, there are two types of clustering methods performed on the usage data: Web transaction clustering and Web page clustering [8]. One successful application of Web page clustering is the adaptive Web site. For example, the algorithm called PageGather [46, 81] is proposed to synthesize index pages that do not exist initially, based on partitioning Web pages into various groups. The generated index pages are conceptually representing the various access interests of users according to their navigational histories. Another example is that clustering user rating results has been successfully adopted in collaborative filtering applications as a data preparing step to improve the scalability of recommendation using *k*-Nearest-Neighbour (kNN) algorithm [76]. Mobasher et al. [29] utilize Web transaction and page clustering techniques, which is employing the traditional *k*-means clustering algorithm to characterize user access patterns for Web personalization based on mining Web usage data. These proposed clustering-based techniques have been proven to be efficient from their experimental results since they are really capable of identifying the intrinsic common attributes revealed from their historic clickstream data. Generally, these usage patterns are explicitly captured at the level of user session or page. They, however, do not reveal the underlying characteristics of user navigational activities as well as Web pages. For example, such discovered usage patterns provide little information of why such Web transactions or Web pages are grouped together, and latent relationships among the co-occurrence observation data have not been incorporated into the mining processes as well. Thus, it is necessary to develop LSA-based approaches that can reveal not only common trends explicitly, but also take the latent information into account implicitly during mining. In [37], an algorithm based on Principal Factor Analysis (PFA) model derived from statistical analysis, is proposed to generate user access patterns and uncover latent factors by clustering user transactions and analysing principal factors involved in the Web usage mining. Analogous, some studies [82-84] are addressed to derive user access patterns and Web page segments from various types of Web data, by utilizing a so-called Probabilistic Semantic Latent Analysis (PLSA) model, which is based on a maximum likelihood principle from statistics.

3.6. Conclusion

In this chapter, we have proposed a LSI-based approach, named LUI, for grouping Web transactions and generating user profiles. Firstly, we model the relationships among the co-occurrence observations (i.e. user sessions) into a usage data model in the form of a session-page matrix. Then, a dimensionality reduction algorithm based on the SVD algorithm has been employed on the usage matrix to capture the latent usage information for partitioning user sessions. Based on the decomposed latent usage information, we propose a *k*-means clustering algorithm to generate user session clusters. Moreover, the discovered user groups are utilized to construct user profiles expressed in the form of a weighted page collection, which represent the common usage pattern associated with one specific user access pattern. The constructed user profiles corresponding to various task-oriented behaviours are represented as a set of page-weight pairs, in which each weight reflects the significance contributed by the page. Experiments have been conducted on two real world datasets to validate the effectiveness and efficiency of the proposed LUI algorithm. Meanwhile, an evaluation metric is adopted to assess the quality of the discovered clusters in comparison with existing clustering algorithms. The experimental results have shown that the proposed approach is capable of effectively discovering user access patterns and revealing the underlying relationships among user visiting records.

4. Discovering Usage Pattern and Latent Factor with Probabilistic Latent Semantic Analysis

4.1. Introduction

In the context of Web usage mining, one important task is to reveal intrinsic user navigational patterns and a latent task space. Such kind of usage knowledge can be discovered by a wide range of statistical methods, machine learning and data mining algorithms. Amongst these techniques, LSA based on a probability inference approach is a promising paradigm which can not only reveal the underlying correlations hidden in Web co-occurrence observations, but also identify the latent task factor associated with usage knowledge.

In this chapter we aim to introduce a *Probabilistic Latent Semantic Analysis* (PLSA) model to generate Web user groups and Web page clusters based on latent usage analysis. The remainder of this chapter is structured as follows: we first introduce the theoretical background of the PLSA model in section 4.2, then propose the algorithms for discovering user access patterns and latent factors based on the PLSA model in section 4.3, experiments and analysis are carried out to demonstrate the effectiveness of the proposed approaches in section 4.4, section 4.4 also presents the experimental result discussions regarding the derived usage knowledge as well as the latent task space. Finally, related work and conclusion of this chapter are given in section 4.5 and 4.6, respectively.

4.2. Probabilistic Latent Semantic Analysis Model

The PLSA model has been firstly presented and successfully applied in text mining by [66]. In contrast to standard LSI algorithms, which utilize the Frobenius norm as an optimization criterion, PLSA model is based on a maximum likelihood principle, which is derived from the uncertainty theory in statistics.

Basically, the PLSA model is based on a statistic model called aspect model, which can be utilized to identify the hidden semantic relationships among general co-occurrence activities. Theoretically, we can conceptually view the user sessions over Web pages space as co-occurrence activities in the context of Web usage mining, to infer the latent usage pattern. Given the aspect model over the user access pattern in the context of Web usage mining, it is first assumed that there is a latent factor space $Z = \left(z_1, z_2, \cdots z_k\right)$, and each co-occurrence observation data (s_i, p_j) (i.e. the visit of a page p_j in a user session s_i) is associated with the factor $z_k \in Z$ by a varying degree to z_k. According to the viewpoint of aspect model, it can be inferred that there do exist different relationships among Web users or pages corresponding to different factors. Furthermore, the different factors can be considered to represent the corresponding user access patterns. For example, during a Web usage mining process on an e-commerce website, we can define that there exist *k* latent factors associated with *k* kinds of navigational behaviour patterns, such as z_1 factor standing for having interests in sports-specific product category, z_2 for sale product interest and z_3 for browsing through a variety of product pages in different categories and z_4 … etc,. In this manner, each co-occurrence observation data (s_i, p_j) may convey user navigational interest by mapping the observation data into a *k*-

dimensional latent factor space. The degree, to which such relationship is “explained” by each factor, is derived by a conditional probability distribution associated with the Web usage data. Thus, the goal of employing the PLSA model is to determine the conditional probability distribution, in turn, to reveal the intrinsic relationships among Web users or pages based on a probability inference approach. In one word, the PLSA model is to model and infer user navigational behaviours in a latent semantic space, and identify the latent factors associated. Before we propose the PLSA based algorithm for Web usage mining, it is necessary to introduce the mathematical background of the PLSA model, and the algorithm which is used to estimate the conditional probability distribution.

Firstly, let’s introduce the following probability definitions:

- $P(s_i)$ denotes the probability that a particular user session s_i will be observed in the occurrence data,
- $P(z_k|s_i)$ denotes a user session-specific probability distribution on the latent factor z_k,
- $P(p_j|z_k)$ denotes the class-conditional probability distribution of pages over a specific latent factor z_k.

Based on these definitions, the probabilistic latent semantic model can be expressed in the following way:

- Select a user session s_i with a probability $P(s_i)$,
- Pick a hidden factor z_k with a probability $P(z_k|s_i)$,
- Generate a page p_j with a probability $P(p_j|z_k)$;

As a result, we can obtain an occurrence probability of an observed pair (s_i, p_j) by adopting the latent factor variable z_k. Translating this process into a probability model results in the expression:

$$P(s_i, p_j) = P(s_i) \cdot P(p_j \mid s_i) \tag{4.1}$$

where,

$$P(p_j \mid s_i) = \sum_{z \in Z} P(p_j \mid z) \cdot P(z \mid s_i) \tag{4.2}$$

By applying Bayesian rule, a re-parameterized version will be transformed based on equations (4.1) and (4.2) as

$$P(s_i, p_j) = \sum_{z \in Z} P(z) \bullet P(s_i \mid z) \bullet P(p_j \mid z) \tag{4.3}$$

Following the likelihood principle, we can determine the total likelihood Li of the observation as

$$Li = \sum_{s_i \in S, p_j \in P} m(s_i, p_j) \cdot \log P(s_i, p_j) \tag{4.4}$$

where $m(s_i, p_j)$ corresponds to the entry of the session-page matrix associated with the session s_i and the page p_j, which is discussed in the Web usage data model in chapter 2. In order to maximize the total likelihood, we need to repeatedly generate the conditional probabilities of $P(z)$, $P(s_i|z)$ and $P(p_j|z)$ by utilizing the usage observation data. Known from statistics, *Expectation Maximization* (EM) algorithm is an efficient procedure to perform maximum likelihood estimations in a latent variable model [85]. Generally, two steps need to be implemented alternately: (1) *Expectation* (E) step where posterior probabilities are calculated for the latent factors based on the current estimates of conditional probability, and (2) *Maximization* (M) step, where the estimated conditional probabilities are updated and used to maximize the likelihood based on the posterior probabilities computed in the previous E step.

We now discuss the whole procedure in details as follows:

Firstly, suppose randomized initial values of $P(z_k)$, $P(s_i \left| z_k \right.)$, $P(p_j \left| z_k \right.)$ are given.

Then, in the E-step, we can simply apply Bayesian formula to generate the following variable based on the usage observation:

$$P(z_k \mid s_i, p_j) = \frac{P(z_k) \bullet P(s_i \mid z_k) \bullet P(p_j \mid z_k)}{\sum_{z_k \in Z} P(z_k) \bullet P(s_i \mid z_k) \bullet P(p_j \mid z_k)} \tag{4.5}$$

Furthermore, in M-step, we can compute:

$$P(p_j \mid z_k) = \frac{\sum_{s_i \in S} m(s_i, p_j) \bullet P(z_k \mid s_i, p_j)}{\sum_{s_i \in S, p_j^{'} \in P} m(s_i, p_j^{'}) \bullet P(z_k \mid s_i, p_j^{'})} \tag{4.6}$$

$$P(s_i \mid z_k) = \frac{\sum_{p_{j\in} \in P} m(s_i, p_j) \bullet P(z_k \mid s_i, p_j)}{\sum_{s_i^{'} \in S, p_j \in P} m(s_i^{'}, p_j) \bullet P(z_k \mid s_i^{'}, p_j)} \tag{4.7}$$

$$P(z_k) = \frac{1}{R} \sum_{s_i \in S, p_j \in P} m(s_i, p_j) \bullet P(z_k \mid s_i, p_j) \tag{4.8}$$

where

$$R = \sum_{s_i \in S, p_j \in P} m(s_{i,}, p_j) \tag{4.9}$$

Basically, substituting equations (4.6)-(4.8) into (4.3) and (4.4) will result in the monotonically increasing of total likelihood L_i of the observation data. The iterative implementation of E-step and M-step is repeating until L_i is converging to a local optimal limit, which means the calculated results can represent the optimal probability estimates of the usage observation data. From the previous formulation, it is easily found that the computational complexity of the PLSA model is $O(mnk)$, where *m*, *n* and *k* denote the numbers of user sessions, Web pages and latent factors, respectively.

By now, we have obtained the conditional probability distribution of $P(z_k)$, $P(s_i|z_k)$ and $P(p_j|z_k)$ by performing E- and M-step iteratively. The estimated probability distribution which is corresponding to the local maximum likelihood *Li* contains the useful information for inferring semantic usage factors, performing Web user session clustering and generating the aggregated user profiles which are described in next sections.

4.3. Constructing User Access Pattern and Identifying Latent Factor with PLSA

As discussed in section 4.2, each latent factor z_k do really represent a specific aspect associated with the usage co-occurrence activities in nature. In other words, for each factor, there might exist a task-oriented user access pattern corresponding to it. We, thus, can utilize the class-conditional probability estimates generated by the PLSA model to produce the aggregated user profiles for characterizing user navigational behaviours. Conceptually, each aggregated user profile will be expressed as a collection of pages, which are accompanied by their corresponding weights indicating the contributions to such user group made by those pages. Furthermore, analysing the generated user profile can lead to revealing common user access interests, such as dominant or secondary "theme" by sorting the page weights.

4.3.1. Partitioning User Sessions

Firstly, we begin with the probabilistic variable $P(s_i|z_k)$, which represents the occurrence probability in the condition of a latent class factor z_k exhibited by a given user session s_i. On the other hand, the probabilistic distribution over the factor space of a specific user session s_i can reflect the specific user access preference over the whole latent factor space, therefore, it may be utilized to uncover the dominant factors by distinguishing the top probability values. Therefore, for each user session s_i, we can further compute a set of probabilities $P(z_k|s_i)$ over the latent factor space via Bayesian formula as follows:

$$P(z_k \mid s_i) = \frac{P(s_i \mid z_k) \bullet P(z_k)}{\sum_{z_k \in Z} P(s_i \mid z_k) \bullet P(z_k)} \tag{4.10}$$

Actually, the set of probabilities $P\left(z_k|s_i\right)$ is tending to be “sparse”, that is, for a given s_i, typically only a few entries are significantly different from the predefined threshold. Hence we can classify the user into corresponding clusters based on these probabilities exceeding the given threshold. Since each user session can be expressed as a pages vector in the original *n*-dimensional space, we can create a mixture representation of the collection of user sessions within same cluster that associated with the factor z_k in terms of a collection of weighted pages. The algorithm for partitioning user sessions is described as following.

[Algorithm 4.1]: Partitioning user session

[Input]: A set of calculated probability values of $P(z_k \left| s_i \right.)$, a user session-page matrix SP_{ij}, and a predefined threshold μ.

[Output]: A set of session clusters $SCL = \left(SCL_1, SCL_2, \cdots SCL_k \right)$

Step 1: Set $SCL_1 = SCL_2 = \cdots = SCL_k = \phi$,

Step 2: For each $s_i \in S$, select $P\left(z_k \left| s_i \right. \right)$, if $P\left(z_k \left| s_i \right. \right) \geq \mu$, then $SCL_k = SCL_k \bigcup s_i$,

Step 3: If there are still users sessions to be clustered, go back to step 2,

Step 4: Output session clusters $SCL = \left\{ SCL_k \right\}$.

[Algorithm 4.2]: Generating user profile

[Input]: A session cluster set $SCL = \left\{ SCL_k \right\}$.

[Output]: A set of user profiles $UP = \left\{ UP_k \right\}$.

Step 1: For each factor z_k, choose all candidate sessions in SCL_k,

Step 2: Represent each session s_i as a page vector and compute their *centroids* in the form of page vector as:

$$UP_k = \frac{\sum_i s_i \bullet P(z_k \mid s_i)}{\left| R \right|} \tag{4.11}$$

where $\left| R \right|$ denotes the total number of sessions in the cluster,

Step 3: if there are still user session clusters not to be processed, go back to step 1,

Step 4: Output the *centroid* of each session cluster in the form of page vector as the aggregated user profile corresponding to each factor z_k.

By now, we assign user sessions into the corresponding clusters which can be considered to represent user navigational patterns based on the calculated conditional probability distributions from the PLSA model and characterize the representations of the user profiles in terms of weighted page vector as well. As discussed above, it can be seen that a particular user session does not only belong to just one cluster, but also to other different clusters associated with different latent factors. For example, a user session may exhibit different interests (with different probabilities) on two aspects z_1 and z_2. This can be "explained" as that a user may, indeed, perform different tasks during the same session and really reflect the nature of user access patterns in real world. It can be implied, in turn, the PLSA model partitions user session-page pairs, which is different from clustering either user sessions or pages or both. In other words, the user session-page probabilities in the PLSA model reflect "overlay" of latent factors, while the conventional clustering model assumes there is just one cluster-specific distribution contributed by all user sessions in the cluster [66].

4.3.2. Characterizing Latent Semantic Factor

As mentioned in the previous section, the core of PLSA model is the latent factor space. From this point of view, how to characterize the factor space or explain the semantic meaning of factors is a crucial issue in PLSA model. Similarly, we can also utilize another obtained conditional probability distribution $P(p_j|z_k)$ by PLSA model to identify the semantic meaning of the latent factor by partitioning Web pages into corresponding categories associated with the latent factors.

For each hidden factor z_k , we may consider that the pages, whose conditional probabilities $P(p_j|z_k)$ are greater than a predefined threshold, can be viewed as providing similar functional components corresponding to the latent factor. In this way, we can select all pages with probabilities exceeding a certain threshold to form an aspect-specific page group. By analysing the URLs of the pages and their weights derived from the conditional probabilities, which are associated with the specific factor, we may characterize and explain the semantic meaning of each factor. In section 4.4, we will present two examples with respect to the discovered latent factors. The algorithm to generate the factor-oriented Web page group is briefly described as follows:

[Algorithm 4.3]: Characterizing latent semantic factor

[Input]: A set of conditional probabilities, $P\left(p_j|z_k\right)$, a predefined threshold μ .

[Output]: A set of latent semantic factors represented by a set of dominant pages.

Step 1: Set $PCL_1 = PCL_2 = \cdots = PCL_k = \phi$,

Step 2: For each z_k, choose all Web pages such that $P\left(p_j|z_k\right) \geq \mu$ and $P(z_k|p_j) \geq \mu$, then construct $PCL_k = p_j \cup PCL_k$,

Step 3: If there are still pages to be classified, go back to step 2,

Step 4: Output $PCL = \left\{PCL_k\right\}$.

4.4. Experimental Results and Discussions

In this section, we present some results regarding Web usage patterns and latent semantic factors obtained by conducting experiments on two selected Web log datasets. We first give the latent semantic factor knowledge mined from two datasets, which is titled by the interpretation of the dominant pages. Some examples of user profiles via partitioning the user sessions and calculating the centroids of the session clusters are presented as well.

4.4.1. Data Sets

The first dataset named KDDCUP for experiments has been described in the previous chapter. Here, we would not repeat the descriptions, but only outline the brief information in terms of dataset size and attribute number. Data filtering technique is used to filter out those user sessions visiting less than 4 pages. After data preparation, it includes 9308 user sessions and 69 pages, where the average session length is 11.88 pages. In this data set, the entries in session-page matrix are determined by the numbers of Web page hits since the number of a user coming back to a specific page is a good measure to reflect the user interest on the page.

The second data set is downloaded from msnbc.com (http://kdd.ics.uci.edu/databases/), which describes the page visits by users who visited msnbc.com on September 28, 1999. Visits are recorded at the level of URL category and in a time order. There are 989818 user sessions with 5.7 visits of pages in average for per user in the original data set. After filtering out the user sessions with low visit frequencies, we have constructed the data set for further analysis, which includes 373229 user sessions and 17 URL categories as well. The 17 categories are "frontpage", "news", "tech", "local", "opinion", "on-air", "misc", "weather", "health", "living", "business", "sports", "summary", "bbs" (bulletin board service), "travel", "msn-news", and "msn-sports". In addition, the visit frequency for each user is taken to calculate the weight of the corresponding usage data. We name this data set as “msnbc” dataset.

By considering the number of Web pages and the contents of the Web site carefully and referring to the selection criteria of factors in [84, 86], we choose $k = 13$ (i.e. 13 factors) for KDDCUP dataset and $k = 6$ for msnbs dataset as the initial parameters used in PLSA implementation.

4.4.2. Examples of Latent Semantic Factors

We conduct the experiments on the datasets to extract the latent factors and generate user profiles. Firstly, we present the examples of the latent factors derived from two real data sets by using the proposed PLSA model.

Table 4-1. Latent factors and their characteristic descriptions from KDDCUP

Factor #	Characteristic title
1	Department_search_results
2	ProductDetailLegwear
3	Vendor_ service
4	Freegift
5	ProductDetailLegcare
6	Shopping_cart
7	Online_shopping
8	Lifestyle_assortment
9	Assortment2
10	Boutique
11	Departmet_replenishment
12	Department_article
13	Home page

Table 4-1 first lists 13 extracted latent factors and their corresponding characteristic descriptions from KDDCUP dataset. And Table 4-2 depicts 3 factor examples selected from the whole factor space in terms of associated page information including page number, probability and description. From this table, it is seen that factor #3 indicates the concerns about vendor service message such as customer service, contact number, payment methods as well as delivery support. The factor #7 describes the specific progress which may include customer login, product order, express checkout and financial information input such steps occurred in an internet shopping scenario, whereas factors #13 actually captures another focus exhibited by Web contents, which reveals the fact that some Web users may pay more attentions to the information regarding department itself.

Table 4-2. Factor examples and their associated page information from KDDCUP

Factor #	Page #	P(p_j|z_k)	Page description
#3	10	0.865	main/vendor\.jhtml
	36	0.035	main/cust_serv\.jhtml
	37	0.021	articles/dpt_contact\.jhtml
	39	0.020	articles/dpt_shipping\.jhtml
	38	0.016	articles/dpt_payment\.jhtml
	41	0.016	articles/dpt_faqs\.jhtml
	40	0.013	articles/dpt_returns\.jhtml
#7	27	0.249	main/login2\.jhtml
	44	0.18	checkout/expresCheckout.jhmt
	32	0.141	main/registration\.jhtml
	65	0.135	main/welcome\.jhtml
	45	0.135	checkout/confirm_order\.jhtml
	42	0.045	account/your_account\.jhtml
	60	0.040	checkout/thankyou\.jhtml
#13	12	0.232	articles/dpt_about\.jhtml
	22	0.127	articles/new_shipping\.jhtml
	13	0.087	articles/dpt_about_mgmtteam
	14	0.058	articles\dpt_about_boardofdirectors
	20	0.058	articles/dpt_affiliate\.jhtml
	16	0.053	articles/dpt_about_careers
	19	0.052	articles/dpt_refer\.jhtml
	23	0.051	articles/new_returns\.jhtml

As for msnbc dataset, it is hard to extract the exact latent factor space as the page information provided is described at a coarser granularity level, i.e. URL category level. Hence we only list two examples of discovered latent factors to illustrate the general usage knowledge hidden in the usage data (shown in Table 4-3). The two extracted factors indicate that factor #1 is associated with all kinds of local information that come from miscellaneous information channels such as bbs, while factor #2 reflects the interests or opinions which are often linked with health, sport as well as technical developments in physical exercises.

Table 4-3. Factor examples from msnbc data set

Factor #	Category #	P(p_j|z_k)	URL category
Factor #1	4 7 6 15 16	0.316 0.313 0.295 0.047 0.029	local misc on-air summary bbs
Facto #2	10 5 13 3	0.299 0.262 0.237 0.203	health opinion msn-sport tech

4.4.3. Examples of User Profiles

Furthermore, we can generate user profiles according to the session-specified conditional probabilities $P(z_k|s_i)$ which represent the common visit interests/access patterns of users within the session group. Generally, the aggregated user access profile is expressed as a collection of Web pages ranked by their associated weights, which can reflect the contributions to the specific profile by the corresponding pages. Table 4-4 presents two examples of the generated user profiles. For each profile, the pages are listed in a page sequence ordered by their associated significances in terms of probabilistic values. It may be inferred that the greater weight a page possesses, the more significant contribution the page exhibits. In other words, it is more likely to be visited by users compared to other pages in the user session group. For example, in Table 4-4, the user profile #7 represents the detailed online-shopping activities, especially occurring in purchasing leg-wear products or fashion clothes, whereas user profile #13 reflects one kind of customers' concern focused on the information with regard to the department store itself. Such explanations of user profiles drawn from Table 4-4 are consistent with the knowledge discovered from Table 4-2.

In addition, from the discovered user profiles, we can further conclude that there are more than one kinds of access interests or exists “overlapping” of visiting tendencies involved in one user profile. But we may still distinguish the dominant or secondary “theme” from the other based on the corresponding weights associated with Web contents. Furthermore, the discovered user access profiles make it feasible to benefit further Web analysis applications, for example, Web recommendation or prediction.

Table 4-4. Example of user access profile

	Page #	Weight	Page description
Profile #7	4	1.20E-4	products/productDetailLegwear
	29	8.44E-5	main/shopping_cart
	27	6.50E-5	main/login2
	8	5.20E-5	main/home
	44	5.03E-5	checkout/expresCheckout
	65	3.86E-5	main/welcome
	2	3.77E-5	main/boutique
	7	3.75E-5	main/search_results
	6	3.75E-5	main/departments
	45	3.72E-5	checkout/confirm_order
	32	3.57E-5	main/registration
	42	1.65E-5	account/your_account
Profile #13	12	1.73E-4	articles/dpt_about
	8	9.77E-5	main/home
	13	7.17E-5	articles/dpt_about_mgmtteam
	4	5.91E-5	products/productDetailLegwear
	14	5.17E-5	articles\dpt_about_boardofdirectors
	2	4.85E-5	main/boutique
	16	4.79E-5	articles/dpt_about_careers
	22	4.55E-5	articles/new_shipping
	17	4.26E-5	articles/dpt_about_investor
	18	4.14E-5	dpt_about_pressreleases
	15	3.83E-5	dpt_about_healthwellness
	20	3.78E-5	articles/dpt_affiliate

4.5. Related Work and Discussion

The Internet has become very popular recently and brought us a powerful platform to disseminate, retrieve information as well as conduct business. Generally, users are usually performing their interest-oriented activities by clicking or visiting one or more functional Web items. In this manner, different click streams of Web pages will be recorded in Web log files. Thus, capturing different Web user access interests or patterns can, not only provide helps for Web site structural improvements, but also for better understanding common user navigational behaviour within the same customer group. This, furthermore, can help to recommend or predict the tailored and personalized Web contents to users, and benefit users obtaining more preferred information and reducing waiting time as well.

In order to characterize access patterns from Web server log files, many Web usage mining techniques have been developed by researchers from a variety of academia areas, and many studies have been published to present their great successes achieved in such fields as Web personalization and recommendation systems [40, 41, 49, 50], Web system improvement [51], Web site modification or redesign [46, 52], and business intelligence and e-commerce [5]. With the benefit of great progress in data mining research community, many data mining techniques, such as Web user or page clustering [29, 45, 46], association rule mining [47, 48] and sequential pattern mining technique [19] are adopted in current Web usage mining methods and have achieved great successes as well.

Latent Semantic Indexing (LSI) model is an approach to capture the latent or hidden semantic relationships among co-occurrence activities [61]. Generally, in order to reveal such deeply latent information, the relationships between co-occurrence objects are first modelled into a high dimensional matrix. Then, a dimensional reducing algorithm such as *Singular Value Decomposition* (SVD) or *Principal Component Analysis* (PCA) algorithm, is applied to perform a linear projection of the original relationship space to generate a dimensionality-reduced latent space [61-63]. LSI has been widely used in Web information indexing and retrieval applications [62, 64].

Although LSI has achieved great successes in some applications, it still has some shortcomings due to its improvable statistical foundation [66]. Probabilistic latent semantic analysis (PLSA) model is a probabilistic variant of LSI just as its name indicates. Although they share similar concept of latent semantic analysis, but there still exists distinct difference between them on the level of theoretical basis. Due to the sound solid foundation, PLSA model may outweigh the conventional LSI algorithms in some particular applications. Recently, approaches based on PLSA model have been successfully applied in collaborative filtering [78] Web usage and content mining [84], text learning and mining [86, 87], co-citation analysis [88] and related topics.

4.6. Conclusion

In this chapter, we have developed a *Probabilistic Latent Semantic Analysis* (PLSA) model, which can infer the hidden semantic factors and uncover user access patterns from the session-page observation data. We started with introducing the theoretical background of PLSA model. The motivation behind of this model is on a basis of an assumption that each co-occurrence observation is associated with a set of latent aspects or tasks, whose degrees could be determined from a probability inference process. By using PLSA model, the latent factor space and user profiles have been successfully revealed by employing algorithms of clustering pages and user sessions based on the estimates of conditional probability distribution. The experiments on two real world data sets have been conducted to evaluate the effectiveness of the proposed method. The experimental results have shown that the latent factors can be textually inferred based on the interpretation of the semantic factor space. In addition, the user access patterns have also been characterized by the user profiles, which are expressed in the forms of weighted page sets. The significance weights of pages in the user profiles can be utilized to determine the main and secondary "theme" of the navigational behaviours, which will provide helps for further Web applications, such as Web recommendation.

5. Web Usage Mining Using Latent Dirichlet Allocation Model

5.1. Introduction

In the previous two chapters, we presented a conventional LSA model and a variant of LSA model for Web usage mining, which results in discovering usage based user session segments and Web page groups. However, the former one is lack of the capability of capturing latent semantic factors and suffers from the problems of sparsity of input data, especially in case of a huge number of Web objects appeared in the constructed usage data, while the latter one often incurs overfitting problems, which is very common in the context of data mining and machine learning. To address these difficulties occurred in the proposed approaches, in this chapter, we aim to explore a new LSA-based paradigm, named *Latent Dirichlet Allocation* (LDA) model, for Web usage mining and Web recommendation.

The remainder of this chapter is structured as follows: section 5.2 discusses the theoretical background of LDA model, and the algorithms for estimating the variational parameters of LDA model and for discovering usage pattern knowledge; in particular, we systematically review the development of generative models, including the algorithms of LSI, PLSA and LDA for latent semantic analysis in an evolution manner, and describe how these techniques are used to deal with the identified research problems. We present the experiments conducted on the collected dataset in section 5.3. Evaluation of LDA model is also given in this section. The related work is reviewed in section 5.4. Finally we conclude this chapter in section 5.5.

5.2. Latent Dirichlet Allocation Model

Prior to presenting the algorithm based on LDA model for Web usage mining and Web recommendation, we first discuss the usage data model and the evolution of generative models, which address the data analytical models used in this study.

5.2.1. Usage Data Matrix

To explore capturing of the inherent property of user access behaviour for predicting a user's access task, here, we briefly review the usage data model in the forms of vector matrix that we intend to manipulate at first. We use the following notations to model the co-occurrence activities of Web users and pages:

- $S = \{s_1, s_2, \cdots s_m\}$: a set of m user sessions.
- $P = \{p_1, p_2, \cdots p_n\}$: a set of n Web pages.
- For each user, the navigational session is represented as a sequence of visited pages with corresponding weights: $s_i = \{a_{i1}, a_{i2}, \cdots a_{in}\}$, where a_{ij} denotes the weight for a page p_j visited in a s_i user session. The corresponding weight is usually determined by the number of hits or the amount time spent on the specific page.
- $SP_{m \times n} = \{a_{ij}\}$: the ultimate usage data in the form of weight matrix with dimensionality of $m \times n$.

Generally, the element in the session-page matrix, a_{ij} , is the normalized weight associated with the page p_j in the user session s_i. The session normalization is able to capture the relative significance of a page within one user session with respect to other pages accessed by same user. For example, Figure 5-1 depicts a usage snapshot from a Web log file and its corresponding usage data in the form of weighted page vector [84]. Particularly, the session begins with a line of the form:

SESSION #n (USER_ID = k)

where *n* is the session number, and *k* is the user id. Within a given session, each line corresponds to one specific page access. Each line in a session is a tab delimited sequence of 3 fields: time stamp, page accessed, and the referrer. The time stamp represents the number of seconds relative to January 1, 2002. To eliminate the influence of the variational visit duration each element in the usage matrix is normalized by calculating the ratio of the visiting time on its corresponding page to total visiting time, e.g.

$w_{ect2002.pdf} = 55/(14+68+326+6+55)*100 = 11.73\ldots$ and so on.

```
SESSION #925  (USER_ID = 338)
9745438     /news/default.asp           -
9745452     /admissions/      /news/default.asp
9745520     /admissions/requirements.asp     /admissions/
9745846     /programs/        /admissions/requirements.asp
9745852     /programs/2002/gradect2002.asp          /programs/
9745907     /pdf/promos/2002/ect2002.pdf    /programs/2002/gradect2002.asp
```

Page url	Duration(s)	weight(%)
/news/default.asp:	14	2.98
/admissions/	68	14.5
/admissions/requirements.asp	326	69.51
/programs/	6	1.28
/programs/2002/gradect2002.asp	55	11.73
/pdf/promos/2002/ect2002.pdf	…	…

Figure 5-1. A usage snapshot and its normalized weight expression

5.2.2. Generative Models

Based on the constructed Web usage data model in the form of weight vector over the page space, we then intend to develop a mechanism to learn the underlying properties of the usage data and extract the informative knowledge of Web access behaviour to model user access patterns. Before we present LDA model for Web usage mining, it is essential to first recall various analytical models used for co-occurrence observations in the context of text mining [89]. Although these data analysis models are initially proposed for revealing intrinsic association between documents and words, it is appropriate and reasonable to introduce the basic idea into our identified research problems, which helps us to easily understand the theoretical bases as well as the strengths of the proposed techniques, for accomplishing the required tasks in the context of Web usage mining.

Currently there are generally two kinds of machine learning techniques that can accomplish the tasks, namely generative and discriminative model. In the generative model, we discover the model of date source through a generating procedure, whereas we learn directly the desired outcome from the training data in the descriptive model. In this study, we will apply the generative model to extract Web usage knowledge.

In particular, here we aim to introduce a recently developed generative model called *Latent Dirichlet allocation* (LDA), and explore how to employ it to model the underlying correlation amongst usage data. LDA is one kind of generative probabilistic models that are able to effectively generate infinitive sequences of samples according to a probability distribution. The probability inference algorithm is then used to capture the aggregate property of user sessions or Web pages associated with the user access patterns, and reveal the semantics of the topic space via referring to the derived distribution of user sessions or page objects over the latent task space implicitly.

The generative model, sometimes called Gaussian mixture model, can be generally used to represent user sessions via a vector expression. In this model, each user session/visit is considered to be generated by a mixture of topics, where each topic is represented by a Gaussian distribution with a mean and variation value. The parameters of mean and variation are estimated by an EM algorithm.

Like language models of information retrieval where words are modelled as the co-occurrence in a document, we intend to formulate user hits or duration spent on different pages as a session-page occurrence. Here each user session that consists of a number of weighted Web pages, is considered to be equivalent to a document whereas the whole Web page set is treated as similarly as a “bag-of-word” concept in text mining.

The simplest probability model in text mining is unigram model, where the probability of each word is independent of other words which have already appeared in the document. This model is considered as a single probability distribution U over an entire vocabulary V, i.e. a vector of probabilities, $U(v)$ for each word v in the vocabulary.

Under the unigram model, words appeared in every document are randomly taken out from a bag-of-word, and then their values are estimated. Therefore, the probability of an observed sequence of words, $w = w_1, w_2 \cdots w_n$ is:

$$P_{uni}(w) = \prod_{i=1}^{n} U(w_i) \tag{4.12}$$

The main limitation of the unigram model is that it assumes all documents are only of homogeneous word collections, that is, all documents exhibit a single topic, which is theoretically modelled as the probability distribution *U*. However, this assumption is often not true in real scenarios, since typically, most documents are in relation to more than one topic, which would be represented by a markedly distinctive set of distributions. Especially, in the context of mining user access pattern, almost every visitor would have different preferences rather than only one intention.

In order to handle the heterogeneous property of documents, mixture model is introduced to overcome the previous problem. In this generative model, we first choose a topic, z, according to a probability distribution, T, and then, based on this topic, we select words according to the probability distribution of the z^{th} topic. Similarly, the probability of a observed sequence of words, $w = w_1, w_2 \cdots w_n$, is formulated as:

$$P_{mix}\left(w\right) = \sum_{z=1}^{k} T(z) \prod_{i=1}^{n} U_z(w_i) \tag{4.13}$$

The main drawback with the mixture model is that it still considers each document to be homogeneous although it could better tackle the heterogeneity in document collections. Similar problem will occur in the context of Web usage mining. It is thus needed to develop a better model to tackle the heterogeneity nature of document collections, i.e. multi-topic distributions of probability.

Probabilistic Latent Semantic Analysis (PLSA) model is an appropriate model that is capable of handling the multi-topic property in the process of Web text or usage mining [66]. In this model, for each word that we observe we pick up a topic according to a distribution, T, which is dependent on the document. The distribution models the mixture of topics for one specific document. And each topic is associated with a probability distribution over the space of the word vocabulary and the document corpus, derived from a generative process. The probability distribution of an observed sequence $w = w_1, w_2 \cdots w_n$ is parameterized as:

$$P_{plsa}(w) = \prod_{i=1}^{n} (\sum_{z=1}^{k} T_{w_i}(z) U_z(w_i)) \tag{4.14}$$

There are two main problems with PLSA model: 1) it is hard to estimate probability distributions of a previously unseen document; 2) due to the linear growth in parameters that depend on the document itself, PLSA suffers from the problems of over-fitting and inappropriate generative semantics.

To address these problems, *Latent Dirichlet Allocation* (LDA) is introduced by combining the basic generative models with a prior probability on topics to provide a complete generative model for documents. The basic idea of LDA is that documents are modelled as random mixtures over latent topics with a probability distribution, where each topic is represented by a distribution over word vocabulary. In this sense, any random mixing distribution, $T(z)$ is determined by an underlying distribution thereby representing an uncertainty over a particular $\theta(\cdot)$ as $p_k(\theta(\cdot))$, where p_k is defined over all $\theta \in P_k$, the set of all possible $(k-l)$ -simplex. That is, the Dirichlet parameters determine the uncertainty, which models the random mixture distribution over semantic topics. The generative model is, therefore, expressed as follows:

- pick a mixing distribution $\theta(\cdot)$ from P_k with a probability $p_k(\theta)$
- for each word
- Choose a topic z with a probability $\theta(z)$.
- Choose a word w_i from the topic z with a probability $T_z(w_i)$.

The probability of observing a sequence of words, $w = w_1, w_2 \cdots w_n$ in this model is:

$$P_{LDA}(w) = \int_{P_k} \left\{ \prod_{i=1}^{n} \sum_{z=1}^{k} \theta(z) T_z(w_i) \right\} p_k(\theta) d\theta \qquad (4.15)$$

where

$$p_k(\theta) = \Gamma\left(\sum_{z=1}^{k} \alpha_z\right) \prod_{z=1}^{k} \frac{\theta(z)^{\alpha_z - 1}}{\Gamma(\alpha_z)} \qquad (4.16)$$

is the Dirichlet distribution with parameters $\alpha_1, \alpha_2 \cdots \alpha_k$. In this model, the ultimate aim is to estimate the parameters of Dirichlet distribution and the parameters for each of the k topic models. Although the integral in this expression is intractable for an exact inference, $T_z(w_i)$ is actually estimated by using a wide range of approximation inference algorithms, such as the variational inference algorithm. The graphical model representation of LDA is illustrated in Figure 5-2.

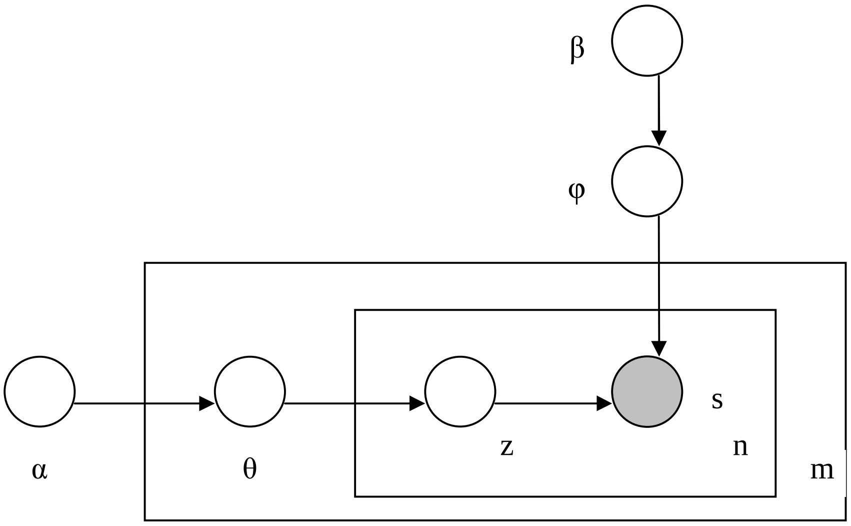


Figure 5-2. Graphical model representation of LDA

5.3. Using LDA for Discovering Access Pattern

Alike capturing the underlying topics over the word vocabulary and each document's probability distributions over the mixing topic space, LDA could also be used to discover hidden access topics (i.e. tasks) and user preference mixtures over the uncovered topic space from the user surfing history. That is, from the usage data, LDA can identify the latent topics in the form of a simplex of Web pages, and characterizes each Web user session as a simplex of these discovered topics. In other words, LDA reveals two aspects of underlying usage information to us, that is, the hidden topic space and the topic mixture distribution of each Web user session, which reflects the underlying correlation between Web pages as well as Web user sessions. With the discovered topic-simplex expression, it is possible to model user access patterns in terms of topic mixture distributions, in turn, to predict user's potentially interested pages by employing a collaborative recommendation algorithm. In the following parts, we discuss how to discover user access patterns in terms of topic-simplex expressions as well as the latent topic space based on LDA model.

Similar to the implementation of the document-topic expression in text mining discussed above, viewing Web user sessions as mixtures of topics makes it possible to formulate the problem of identifying the underlying topics/tasks hidden in the usage data. Given m Web user sessions expressing z topics over n distinctive pages, we can represent $P(p|z)$ with a set of z multinomial distributions ϕ over the n pages, such that $P(p|z=j)=\phi_p^{(j)}$, and $P(z)$ with a set of m multinomial distributions θ over the z topics, such that for a page in Web session *s*, $P(z=j)=\theta_j^{(s)}$. To discover the set of topics hidden in a collection of Web pages $p = \{p_1, p_2, \dots p_n\}$, where each p_i appears in some Web sessions, our aim is to obtain an estimate of ϕ that gives a high probability over the pages in the page collection. Here we use LDA model described above to estimate ϕ and θ parameters that result in a maximum log likelihood of the usage data. The complete probability model is as follows:

$$\begin{aligned} &\theta \sim Dirichlet(\alpha) \\ &z_i \left| \theta^{s_i} \sim Discrete(\theta^{s_i}) \right. \\ &\phi \sim Dirichlet(\beta) \\ &p_j \left| z_i, \phi^{z_i} \sim Discrete(\phi^{z_i}) \right. \end{aligned} \tag{4.17}$$

Here, z stands for a set of hidden topics, θ^{s_i} denotes a Web session s_i 's preference distribution over the topics and ϕ^{z_i} represents the specific topic z_i 's association distribution over the page collection. α and β are hyperparameters of the prior of θ and ϕ. In this manner, the equation (4.15) is re-parameterized as

$$P(s_i \left| \alpha, \beta \right.) = \int p(\theta \left| \alpha \right.)(\prod_{j=1}^{n} \sum_{z_k \in Z} p(z_k \left| \theta \right.) p(p_j \left| z_k, \beta \right.))d\theta \tag{4.18}$$

where s_i denotes a user session, n is the number of pages. More details regarding the formulation is referred to [67].

We use a variational inference algorithm to estimate each Web session's correlation with multiple topics (α), and the associations between the topics and Web pages (β), with which we can capture user visit preference distribution exhibited by each Web session and identify the semantics of topic space.

Given a collection of user sessions, we aim to estimate the parameters of α and β that maximize the log likelihood of the usage data

$$li(\alpha,\beta)=\sum_{i=1}^{m}\log P(s_i\,|\alpha,\beta) \tag{4.19}$$

where *m* is the number of user sessions.

The variational EM algorithm [67] executes as follows. E-step updates the optimizing values of the parameters and re-calculates the posterior value of the equation (4.18); M-step maximizes the log likelihood with respect to the updated parameters. This iterative execution results in finding parameters of α and β that correspond to a maximum likelihood of the usage data.

Interpreting the contents of prominent pages related to each topic based on β will eventually result in defining the meaning of each topic. Meanwhile, the topic-oriented user access patterns are constructed by examining the calculated user session's association with multiple topics and aggregating all sessions whose associations with a specific topic are greater than a threshold. We describe our approach to discovering the topic-oriented access pattern below.

Given this representation, for each latent topic, we can consider user sessions with θ^{s_i} exceeding a threshold as "prototypical" user sessions associated with that topic. In other words, these top user sessions that contribute significantly to this topic via their navigational behaviour, are used to construct this topic-specific user access pattern.

Thus, for each latent topic, we choose all user sessions with θ^{s_i} exceeding a certain threshold as candidates of this specific access pattern. As a user session is represented by a weighted page vector in the original space of page collections, we can create an aggregate collection of user sessions to represent this topic-specific access pattern in the form of weighted page vector. The algorithm to generate the topic-specific access pattern is described as follows:

[Algorithm 5.1]: Building user access pattern based on LDA model

[Input]: The calculated session-topic preference distribution θ, the usage data SP and a predefined threshold μ.

[Output]: A set of user access patterns $AP = \{ap_k\}$.

Step 1: For each latent topic z_j, choose all user sessions with $\theta_{z_j}^s \geq \mu$ to construct a user session aggregation R_j corresponding to z_j,

Step 2: For each latent topic z_j, compute the topic-specific aggregated user access pattern of the selected user sessions in R by taking the discovered sessions' associations θ with z_j into account

$$ap_j = \frac{\sum_{s \in R_k} \theta_{z_j}^s \bullet s}{\left|R_j\right|} \qquad (4.20)$$

where $\left|R_j\right|$ is the number of the selected user sessions in R_j,

Step 3: Output a set of topic-oriented user access patterns AP over k multiple topics, $AP = \left\{ap_1, ap_2, \ldots ap_k\right\}$.

5.4. Experiments and Results

In order to evaluate the effectiveness of the proposed LDA model for topic-oriented user access pattern mining, we conduct several experiments on real Web log dataset. Firstly, we employ LDA to extract semantics of topics via interpreting the contents of predominant pages, which possess the estimated probabilities exceeding a predefined threshold, and then, discover the topic-oriented user access pattern using the algorithm 5.1 discussed above. Meanwhile, the quality of the user access pattern is assessed by employing the evaluation metric of clustering quality analysis. Eventually, we carry out comparison against various generative models to demonstrate the effectiveness of the proposed analytical model.

5.4.1. Dataset

The Web log dataset used in this chapter is the same as the one used in the previous chapters. In brief, after data pre-processing stage, the filtered data contains a user visit log file with 13745 user sessions and a Web page collection of 683 pages. By referring to the previous analysis and taking the analysis on the sensitivity of the involved Web content into account, we eventually choose a latent task space with 30 topics as the initial input parameter that used in LDA model.

5.4.2 Evaluation Metric for User Access Pattern

In the context of Web usage mining, one important measure to evaluate the effectiveness of the proposed approach is the quality of user session clusters, derived from above discussed approach. Here, we adopt one previously used metric for evaluating the cluster quality, named *Weighted Average Visit Percentage* (WAVP), which is a commonly used parameter in applications of Web usage mining [29]. This evaluation method is to assess the quality of each user profile individually by computing how likely a user session which contains any pages in the transaction cluster will include the rest pages in the cluster during the same session. In other words, the higher parameter the value is, the better the quality of the cluster is.

5.4.3 Samples of Topics and User Access Preference Distributions

As stated in section 5.3, we can use LDA to identify the semantics of latent topics from the contents of prominent pages contributing significantly to each topic. We first present two examples of topics out of 30 discovered topics in Table 5-1. To illustrate theses topics, we also list the URLs of the prominent pages along with their corresponding probabilities (based on ϕ) respectively. Meanwhile, the estimate of each user session's association with multiple topics (θ) could be used to model each user's navigational preference over the topic space. Figure 5-3 depicts the navigational preference distribution of one specific user session illustrated in Figure 5-1 over the predefined 30 topics.

Table 5-1 indicates two topic examples out of a 30-topic set discovered from the CTI dataset based on β derived by LDA model. By interpreting the URL contents of the predominant pages with probabilities exceeding 1%, it is known that the topic #1 is in relation to the activities of searching information with respect to Master degrees in disciplines of IS or MIS, such as admission and course syllabus etc, whereas the topic #18 is referred to common access interests on browsing associated pages regarding Postgraduate and PhD program, course costs, application of assistantship as well as related faculty and syllabus information. Furthermore, the associated probability of each URL of the specific topic reveals the significance contributed by each corresponding page.

Figure 5-3 illustrates the navigational preference distribution of the user session described in Figure 5-1 over the topic space based on θ. From the figure, it seems that the topic # 16, #28, and #1 dominate the navigational preference of this specific user session. Moreover, we could further infer that this user session is more related to requiring wanted information regarding applying for admission in postgraduate programs related to business course, e.g. ECT course, and the course syllabus information associated with those programs via examining the semantics of all involved topics, e.g. topic # 16 being about whole admission application procedure, #28 being about syllabus information search and the description of topic #1 (displayed in Table 5-1). In this occasion, this user session could be used to mainly contribute the construction of the user access pattern #16, which is corresponding to topic #16 for Web recommendation.

Table 5-1**.** Examples of two topics discovered from CTI dataset

Topic #1		
Page #	URL	Probability
1	/admissions/	0.332
590	/programs/2002/gradis2002.asp	0.227
591	/programs/2002/gradmis2002.asp	0.115
259	/courses/syllabus.asp?course=584-97-301&q=3&y=2002&id=653	0.084
390	/news/news.asp	0.075
414	/pdf/promos/2002/is2002.pdf	0.029
594	/programs/2002/maat2002.asp	0.026
575	/programs/	0.018
666	/programs/masters.asp	0.017
Topic #18		
4	/admissions/costs.asp	0.287
593	/programs/2002/gradtc2002.asp	0.146
592	/programs/2002/gradse2002.asp	0.134
355	/cti/gradassist/assistsubmit.asp	0.110
196	/courses/syllabus.asp?course=450-96-305&q=3&y=2002&id=273	0.037
464	/people/facultyinfo.asp?id=216	0.036
577	/programs/2001/phdincs2001.asp	0.032
605	/programs/courses.asp?depcode=21&deptmne=csc&courseid=211	0.027
248	/courses/syllabus.asp?course=566-98-301&q=3&y=2002&id=302	0.025
354	/cti/gradassist/assistantship_form.asp?section=news	0.024
247	/courses/syllabus.asp?course=566-98-301&q=3&y=2002&id=302	0.024
597	/programs/bulletin.asp	0.019
385	/hyperlink/hyperspring2002/lobby.asp	0.016
249	/courses/syllabus.asp?course=567-98-302&q=3&y=2002&id=423	0.015
641	/programs/courses.asp?depcode=98&deptmne=tdc&courseid=463	0.015

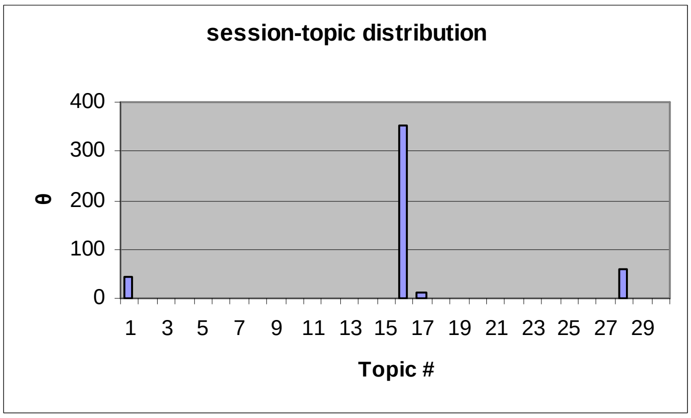


Figure 5-3**.** The specific user session-topic task distribution over the discovered topics

5.4.4. User Access Pattern Evaluation Using Clustering Quality Metric

To investigate the effectiveness of the proposed LDA approach for mining usage pattern from Web log files, evaluation analysis has been carried out to compare the proposed technique with other two models used in the previous chapters. Here we make the comparison by using the metric discussed in section 5.4.2. Figure 5-4 depicts the comparison results in terms of the WAVP metric. From the figure, it is shown that both of PLSA-based and LDA-based techniques are able to achieve better results of cluster quality in comparison with clustering alone approach, especially in the ranges of high WAVP values. For example, for the first five best clusters, the WAVP values derived from the first two techniques are almost 50% higher than that derived by using clustering alone method. When the cluster number increases, the clustering performances of three models trends to be closer and closer, eventually merge together nearly. This observation is becoming significant when the number of clusters reaches the values of 10 or more. Therefore, we can draw the conclusion that the latent semantic analysis paradigms are capable of discovering the underlying linkage between Web user sessions and generating better quality clusters in comparison with the conventional clustering-based method. Furthermore, from two curves of the LSA based models, it is seen that these two kinds of latent semantic analysis have very similar capability of efficiently partitioning Web user sessions. However, it seems that LDA-based approach outperforms the PLSA-based model a bit more in terms of WAVP, especially in the range of the very first clusters. This observation could also be justified by the experimental results in terms of recommendation precision, which are presented in the following chapter 7.

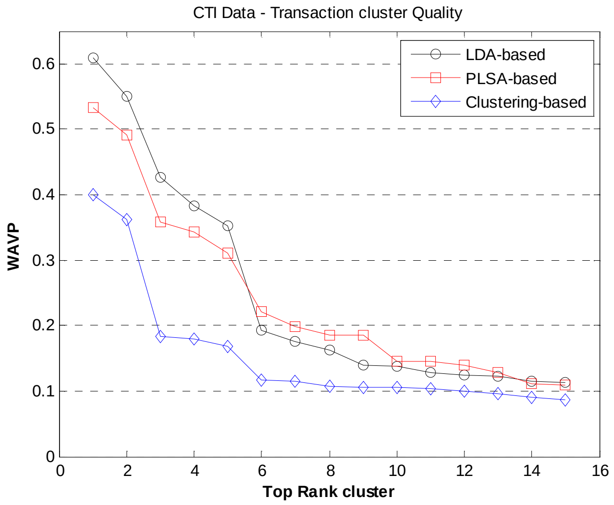


Figure 5-4. Cluster quality comparison of three methods in terms of WAVP metric

5.5. Related work

In the past decades, using latent semantic analysis models for characterizing documents and Web transactions has become a promising trend in machine learning and data mining. The first well-known LSA model, namely *Latent Semantic Indexing* (LSI), was firstly introduced by Deerwester in 1990. The main idea of LSI is to map the high-dimensional input space to a reduced-dimensional feature space, called latent semantic space that captures the underlying associations between the observed co-occurrence observations, for example, the semantic relations between word and document, or Web page and Web user session. The mapping is usually carried out by implementing a Singular Value Decomposition (SVD) operation, which is to extract the maximally approximate matrix to the original input space. LSI assumes there are *k* underlying latent topics, over which documents or Web sessions are associated with accordingly. Those latent topics are, in turn, conceptually considered as document classes or Web session categories, in some extents, which leads to document categorization or Web session clustering.

Based on a more solid statistical base, Hoffman [66] proposed an alternative to LSI, by introducing *Probabilistic Latent Semantic Analysis* model (PLSA), which is able to discover the latent variables from the co-occurrence observations. The model is described as an aspect model, which assumes the existence of a set of unseen factors hidden in the co-occurrences between two sets of objects, like documents/words or Web sessions/pages. This model is essentially a latent statistical mixture model, that is, different words within a document contribute to a variety of topics with various degrees. To tackle PLSA model for revealing the underlying relations, a specific *Expectation-Maximization* (EM) algorithm is employed to maximize the likelihood of the data. The estimation is done in an iterative running manner until the likelihood converges to a local optimal value. Due to its capability and flexibility, PLSA has achieved a great amount of successes in many research studies. Hoffman originally proposed it for text mining and collaborative filtering, while Zhou et al extended this model for Web usage mining by treating Web log data as observed co-occurrences [37]. In [90], Song et al used it for disambiguating author names within a large document collection. They first employed it to extract the topic distribution with respect to persona and words, and then, incorporated a hierarchical agglomerative clustering algorithm to disambiguate authors. Additionally, PLSA is also successfully introduced in personalized Web search and Web service composition [91] studies.

A main shortcoming of PLSA is the overfitting problem, resulted from the fact that the estimated parameters in this model grow linearly with the size of the document collection or Web transaction database, which is often occurred in machine learning procedures. Blei at al. [67] later introduced a Bayesian hierarchical model, named *Latent Dirichlet Allocation* (LDA) for modelling underlying relations in the context of document-topic analysis, in which each document has a distinctive topic distribution, drawn from a conjugate Dirichlet prior that remains unchanged for all documents in a collection. The words within that document are then expressed over the topic space discovered. The model training and parameter estimation can be performed efficiently by employing a variational EM algorithm [67]. In contrast to PLSA and other generative models, LDA is considered as a full generative aspect model, which has a better generalization performance. Many studies have been extensively carried out to employ LDA model in other applications. Wei et al. [92] proposed LDA-based models for documental retrieval by performing LDA analysis in an ad-hoc retrieval task. Li et al. [93] successfully addressed the problems of nature scene classifications to improve the capability of computer vision with this Bayesian hierarchical model. And Wang et al utilized it for identifying topical trends through a time series analysis [94].

5.6. Conclusion

In this chapter, we have proposed a novel Web usage mining approach based on Latent Dirichlet Allocation (LDA) model. With LDA model, the associations between user sessions and navigational topics and the associations between topics and Web page collection hidden in user click log files are discovered via a model generating process. Interpreting the predominant Web pages with the significant contributed probabilities results in revealing the semantics of the underlying topic space, and examining the association between user sessions and multiple topics leads to discovering the user access preference distribution over the topic space, in turn, provides a better way of identifying various common access patterns by aggregating user sessions with similar access preference. The experimental results on the selected usage data set have shown that the proposed LDA-based Web usage mining is capable of revealing the latent task space and generating the user session clusters with better quality in comparison with other conventional LSA based approaches.

6. Discovering Task-Oriented Navigational Distribution for Web Recommendation

6.1. Introduction

Web recommendation is to present more preferable Web contents to Web users by predicting user's navigational preferences from his/her very first clicks on Web pages. The recommended Web content is usually determined by discovering user access preferences and referring to the historic visiting preferences of other users with the similar access pattern via a collaborative filtering technique. Particularly, we address incorporating the usage pattern knowledge discovered by Web usage mining methods that discussed in the previous chapters, with a variant of *K-Nearest Neighbouring* (kNN) algorithm, which is one of the most commonly used recommendation algorithms in machine learning, to build up a Web recommendation framework in the following sections. The proposed recommendation scoring algorithm is called *Top-N weighted scoring scheme* (TopN-WSS). The schematic recommendation framework is outlined as follows: we first utilize the probability estimates discovered from Web usage mining process to extract the latent task/factor space and construct user access patterns (i.e. user profiling). This step can be accomplished offline and results in a usage knowledge base for Web recommendation. When a new user comes to visit a Web site, we predict the user's task preference and recommend the customized Web pages to the user by referring to the discovered usage knowledge base. Based on this framework, we propose several algorithms to address Web recommendation. In the following chapters, we propose two types of recommendation approaches. This chapter is focusing on making recommendation by discovering task-oriented navigational distributions, which is determined by PLSA model, while chapter 7 will discuss a unified user profiling algorithm that could be used with PLSA and LDA model for recommendation.

To implement the algorithm proposed in this chapter, we first identify the user's dominant task sequence from the probability estimates via a Bayesian updating approach. Then, incorporating the identified dominant tasks with the weights of pages, which corresponds to the captured dominant factors, into a recommendation scoring scheme, results in generating a top-N recommendation page set. We, therefore, predict the most likely visited Web pages and recommend them to Web users.

This chapter is structured as follows: we first introduce the proposed TopN-WSS algorithm in section 6.2. Then in section 6.3, we present a Web recommendation algorithm based on discovering user task-oriented navigational distributions over the task space and incorporating the navigational distributions into the TopN-WSS algorithm. Eventually, we present some experimental results in section 6.4 to evaluate the effectiveness and efficiency of the proposed approach. This chapter is concluded in section 6.5.

6.2. Top-N Weighted Scoring Scheme for Web Recommendation

Before we propose the Web recommendation algorithm, it is essential to describe the theoretical background of the recommendation scoring scheme used in the later sections, which is considered as the core point of Web recommendation algorithms. In this study, we introduce a unified recommendation scoring algorithm, named *Top-N Weighted Scoring Scheme* (TopN-WSS), which is a variant of the well known KNN algorithm. In the Top-N WSS scheme, usage information derived from Web mining process is first used to represent the user profiles, which are modelled as vectors of weighted page sequences. When a new user comes in, we then determine the task preference of the user and predict the potentially visited pages based on the page weights in the determined usage pattern via collaborative referring techniques.

The procedure of the TopN-WSS algorithm is described as follows:

[Algorithm 6.1]: Top-N weighted scoring scheme for Web recommendation

[Input]: An active user session s_a and a usage knowledge base.

[Output]: A set of Web pages in a descending order of weights.

Step 1: Given an active user session s_a and the usage pattern knowledge discovered by Web usage mining processes,

Step 2: Capture the access task preference of the current user by matching the current user session with the discovered usage pattern knowledge via various learning algorithms,

Step 3: Calculate the recommendation scores of pages within the selected usage pattern based on the derived page weights,

Step 4: Re-sort the recommendation scores in a descending order, where each score stands for the possibility of page being potentially visited by the current user in next steps, and select the top-N pages with the N highest recommendation scores as the recommendation list of pages.

In the following sections, we aim to combine the usage knowledge discovered by various Web usage mining techniques, into the top-N weighted scoring scheme to produce predictions of pages that are potentially visited.

6.3. Identifying Task-Oriented Navigational Distribution for Web Recommendation with PLSA Model

As we discussed before, each latent factor z_k do really represent a specific aspect associated with the co-occurrence observations in nature. In this sense, we can utilize the factor-conditional probability estimates generated by PLSA model to partition Web pages and induce latent factors by extracting the contents of "dominant" Web pages whose probabilities are exceeding a predefined threshold.

6.3.1. Characterizing Latent Factor Space

First, we discuss how to capture the latent factors associated with user navigational behaviours. This aim is accomplished by characterizing the "dominant" pages. Note that $P(p_j \mid z_k)$ represents the conditional occurrence probability over the page space corresponding to a specific factor, whereas $P(z_k \mid p_j)$ represents the conditional probability distribution over the factor space corresponding to a specific page, which is expressed in the form of

$$P(z_k \mid p_j) = \frac{P(p_j \mid z_k) \cdot P(z_k)}{\sum_{z_k \in Z} P(p_j \mid z_k) \cdot P(z_k)} \tag{6.1}$$

In such an expression, we may consider that the pages whose conditional probabilities $P(p_j \mid z_k)$ and $P(z_k \mid p_j)$ are both greater than a predefined threshold μ can be viewed to contribute significantly to one particular functionality related to the latent factor. Furthermore, we choose all pages satisfying the aforementioned condition to form a number of "dominant" page sets. By exploring the contents of these pages, we may characterize the semantic meaning of each factor. In section 6.4, we will present some examples of latent factors derived from two real data sets. The algorithm to characterize the task-oriented semantic latent factor is described as follows:

[Algorithm 6.2]: Characterize Latent Factors

[Input]: A set of probability estimates $P(p_j \mid z_k)$ and $P(z_k \mid p_j)$, a predefined threshold μ.

[Output]: A set of characteristic page base sets $LF = (LF_1, LF_2, \cdots, LF_k)$.

```
Step 1: $LF_1 = LF_2 = \cdots = LF_k = \phi$,
Step 2: For each $z_k$, choose all pages $p_j \in P$,
    If $P(p_j \mid z_k) \geq \mu$ and $P(z_k \mid p_j) \geq \mu$ then
        $LF_k = LF_k \cup p_j$
    Else go back to step 2,
Step 3: If there are still pages to be classified, go back to step 2,
Step 4: Output $LF = \{LF_k\}$.
```

6.3.2. Identifying Web Page Category

Note that the set of $P(z_k \mid p_j)$ is conceptually representing the probability distribution over the latent factor space for a specific Web page p_j, we, thus, construct the page-factor matrix based on the calculated probability estimates, to reflect the relationships between Web pages and latent factors, which is expressed as follows:

$$vp_j = (c_{j,1}, c_{j,2}, ..., c_{j,k}) \tag{6.2}$$

Where $c_{j,s}$ is the occurrence probability estimate of the page p_j on a factor z_s. In this way, the distance between two page vectors may reflect the functionality similarity exhibited by them. We, therefore, define the similarity by applying the well-known cosine similarity as:

$$sim(p_i, p_j) = \left(vp_i, vp_j\right) \big/ (\|vp_i\|_2 \cdot \|vp_j\|_2) \tag{6.3}$$

where $\left(vp_i, vp_j\right) = \sum_{m=1}^{k} c_{i,m} c_{j,m}$, $\|vp_i\|_2 = \sqrt{\sum_{l=1}^{k} C_{i,l}^2}$

With the defined page similarity measure (6.3), we propose a clustering algorithm to partition Web pages into various page categories. The Web page clustering algorithm is described as follows:

[Algorithm 6.3]: Clustering Web pages

[Input]: A set of $P(z_k \mid p_j)$, a predefined threshold μ.

[Output]: A set of Web page categories $PCL = \{PCL_1, \cdots, PCL_P\}$ and the corresponding centroids $Cid = \{Cid_1, \ldots, Cid_p\}$.

Step 1: Select the first page p_1 as the initial cluster $PCL_1 = \{p_1\}$ and the centroid of this cluster $Cid_1 = p_1$,

Step 2: For each page p_j, measure the similarity between p_j and the centroid of each existing cluster $sim(p_j, Cid_i)$, if $sim\left(p_j, Cid_t\right) = \max_i(sim(p_j, Cid_i)) > \mu$, then insert p_j into the cluster PCL_t and update the centroid of PCL_t as

$$PCL_t = PCL_t \cup p_j \tag{6.4}$$

$$Cid_t = 1/\left|PCL_t\right| \cdot \sum_{j \in PCL_t} vp_j \tag{6.5}$$

where $\left|PCL_t\right|$ is the number of sessions in the cluster PCL_t.

Otherwise, p_j will create a new cluster itself and is the centroid of the new cluster,

Step 3: If there are still sessions to be classified into one of existing clusters or a session that itself is a cluster, go back to step 2 iteratively until it converges (i.e. all clusters' centroids are no longer changed),

Step 4: Output $PCL = \{PCL_i\}$ and $Cid = \{Cid_i\}$.

6.3.3. Web Recommendation Based on Identifying Task Distribution

Generally, Web recommendation process is to predict the customized Web contents to users according to the navigational interests exhibited by individual or groups of users. Suppose that the conditional probabilities are estimated by PLSA model as described above, we can, in turn, utilize them to identify the user's underlying access interest or task and recommend the potentially interested Web presentation to users.

Since each user session is represented as a sequence of visited pages along with different weights, which are determined by the degrees of the interests on these pages of the user, we can capture the interest-oriented task sequence derived from the clicked pages within the session accordingly. This aim is accomplished by computing the posterior probability of each task based on a Bayesian updating approach, given that pages are independent on tasks each other. These posterior probabilities associated with the various tasks indicate the likelihood of the user's underlying intention. The navigational preference, therefore, is characterized as a sequence of tasks with corresponding probabilities. By presetting an appropriate threshold, we can choose all tasks whose posterior probabilities are greater than the defined value as a collection of the dominant tasks to reflect the user's initial intention. Moreover, incorporating the identified sequence of dominant tasks with the task-based page categories derived from the previous section will lead to discovering the page candidates that are more likely to be visited or interested by the user later. The algorithm is described as follows.

[Algorithm 6.4]: Discovering task-oriented navigational distribution for Web recommendation based on PLSA model

[Input]: An active user session $s_i = (p_1^i, p_2^i, \cdots, p_t^i), p_j^i \in P$, a set of estimated conditional probabilities $P(p_j \mid z_k)$ and a threshold μ.

[Output]: The dominant task sequence $TL = \{z_1^i, \cdots, z_t^i\}$ corresponding to the user session and the top-N recommendation pages $RS = \{p_j^r \mid p_j^r \in P, j = 1, 2, \ldots, N\}$.

Step 1: For each task $z_k \in Z$, which is supposed to be independent on the pages, calculate the posterior probability of z_k given all pages in s_i by employing a Bayesian updating method [95]:

$$P(z_k \mid s_i) = \alpha P(z_k) \prod_{p_j^i \in S_i} P(p_j^i \mid z_k) \tag{6.6}$$

where α is a constant,

Step 2: Choose all tasks whose conditional probabilities are greater than the preset threshold to form the dominant task sequence corresponding to the user session.

$$TL = \{z_k \mid z_k \in Z, P(z_k \mid s_i) > \mu\} \tag{6.7}$$

Step 3: For each z_k in *TL*, incorporate the corresponding task-based page category, and then compute the recommendation score for each page p_j as

$$rs(p_j) = \sqrt{\sum_{z_k, p_j} P(z_k \mid s_i) \cdot wt(p_j, z_k)}, p_j \in P, z_k \in TL \tag{6.8}$$

where $wt(p_j, z_k)$ denotes the weight of p_j within z_k page category. Note that the recommendation score will be 0 if the page is already visited in the current session,

Step 4: Sort the computed recommendation scores from step 3 in a descending order, i.e. $rs = (rs(p_1^r), \cdots, rs(p_n^r))$, and choose the N pages with the highest scores to construct the top-N recommendation set.

$$RS = \{p_j^r \mid rs(p_j^r) > rs(p_{J+1}^r), j = 1, 2, \cdots, N-1\} \tag{6.9}$$

6.4. Experiments and Evaluations

In order to evaluate the effectiveness of the proposed Web recommendation method and demonstrate the capability of latent semantic analysis based on PLSA model, we conduct experiments on two real world data sets and summarize the results in the following section.

6.4.1. Data Sets

The first data set we used is downloaded from KDDCUP2000 website. After data preparation, we have setup a usage data set including 9308 user sessions and 69 pages, where each session consists of 11.88 pages in average. We refer to this data set as “KDDCUP data”. In this data set, the numbers of Web page hits by the given user determine the elements in session-page matrix associated with the specific page in the given session.

The second data set is from a academic Website log files [8]. The data is based on a 2-week Web log file during April of 2002. After data pre-processing stage, the filtered data contains 13745 sessions and 683 pages. The entries in the table correspond to the amount of time (in seconds) spent on pages during a given session. For convenience, we refer to this data as “CTI data”. We make use of the usage knowledge derived by PLSA model for Web recommendation.

6.4.2. Latent Task Space

We conduct the experiments on the two data sets to extract the latent factors and group Web pages. Firstly, we present the experimental results regarding the derived latent factors from two real data sets based on PLSA model. Table 6-1 tubulates the results extracted from KDDCUP data set along with the dominant page collection, whereas Table 6-2 presents the results from CTI data set.

From these tables, it is shown that the descriptive titles of latent factors (or tasks) are characterized by some “prominent” pages, whose probabilistic weights are exceeding one predefined threshold. This work is done by interpreting the contents of corresponding pages since these “dominant” pages contribute greatly to the latent factors. With the derived characteristic factors, we may semantically discover the usage-based task patterns.

Table 6-1. Titles of tasks from KDDCUP

Factor	Title	Dominant Page #
1	Department search	6, 7
2	Product information of Legwear	4
3	Vendor service info	10,36,37,39
4	Freegift, especially legcare	1,9,33
5	Product information of Legcare	5
6	Online shopping process	27,29,32,42,44,45,60
7	Assortment of various lifestyle	3,26
8	Vendor2’s Assortment	11,34
9	Boutique	2
10	Replenishment of Department	6,25,26,30
11	Article regarding Department	12,13,22,23
12	Home page	8,35

Table 6-2. Titles of tasks from CTI

Factor	Title	Factor	title
1	specific syllabi	11	international_study
2	grad_app_process	12	Faculty-search
3	grad_assist_app	13	postgrad_program
4	admission	14	UG_scholarship
5	advising	15	tutoring_gradassist
6	program_bacholer	16	Mycti_stud_profile
7	syllabi list	17	schedule
8	course info	18	CS_PhD_research
9	jobs	19	specific news
10	calendar	20	Home page

6.4.3. Examples of Web Page Categories

At this stage, we utilize the aforementioned clustering algorithm to partition Web pages into various page clusters. By analysing the discovered clusters, we may conclude that many groups do really reflect the single user access task; whereas others may span two or more tasks, which may be relevant in nature. As indicated above, the former can be considered to correspond to an intuitive latent factor, and the latter may reveal the "overlapping" property on contents among Web pages.

In Table 6-3, we list three Web page groups out of the total generated groups from KDDCUP data set, which are expressed by the top ranked page information such as page numbers and their corresponding content labels. It is seen that each of these three page groups reflects a sole usage task, which is consistent with the corresponding factor depicted in Table 6-1. Table 6-4 illustrates two Web page groups from CTI data set accordingly. In this table, the upper row lists the top ranked pages and their corresponding contents from the generated page clusters, which reflect the task regarding searching postgraduate program information, and it is easily to conclude that these pages are all contributed to factor #13 displayed in Table 6-2. On the other hand, the significative pages listed in lower row of the table indicate the "overlapping" of two dominant tasks, which are corresponding to factor #3 and #15 depicted in Table 6-2.

Note that with these generated Web page categories; we may make use of these intrinsic relationships among Web pages to improve Web organization designs. For example, an instrumental and suggestive task list based on the discovered page groups can be added into the original Web pages as the means of *Adaptive Web Site Design*, to provide better services to users.

Table 6-3. Examples of Web page groups from KDDCUP

Page	Content	Page	Content
10	main/vendor	38	articles/dpt_payment
28	articles/dpt_privacy	39	articles/dpt_shipping
37	articles/dpt_contact	40	articles/dpt_returns
27	main/login2	50	account/past_orders
32	main/registration	52	account/credit_info
42	account/your_account	60	checkout/thankyou
44	checkout/expresCheckout	64	account/create_credit
45	checkout/confirm_order	65	main/welcome
47	account/address	66	account/edit_credit
12	dpt_about	20	dpt_affiliate
13	dpt_about_mgmtteam	21	new_security
14	dpt_about_boarddirectors	22	new_shipping
15	dpt_about_healthwellness	23	new_returns
16	dpt_about_careers	24	dpt_terms
17	dpt_about_investor	57	dpt_about_the_press
18	dpt_about_pressrelease	58	dpt_about_advisoryboard
19	dpt_refer		

Table 6-4. Examples of Web page groups from CTI

Page	Content	Page	Content
386	/News	588	/Prog/2002/Gradect2002
575	/Programs	590	/Prog/2002/Gradis2002
586	/Prog/2002/Gradcs2002	591	/Prog/2002/Gradmis2002
587	/Prog/2002/Gradds2002	592	/Prog/2002/Gradse2002
65 70 352 353 355	/course/internship /course/studyabroad /cti/…/applicant_login /cti/…/assistantship_form /cti/…/assistsubmit	406 666 678 679	/pdf/forms/assistantship /program/master /resource/default /resource/tutoring

6.4.4. Examples of Task-Oriented Usage Patterns

As described in section 6.3.3, we exploit the posterior probability derived from PLSA model to identify the task-oriented navigational distribution and predict the potentially visited Web pages by combining the task-oriented page categories into the TopN-WSS recommendation process. In the following Table 6-5, we give two examples of the derived navigational task distributions through employing the algorithm 6.4 on two real user sessions from KDDCUP and CTI dataset respectively. In particular, we list the active user sessions along with the corresponding navigational task distribution over the factor space.

From the table, it is easily found that the two users have visited 10 and 11 pages respectively during their browsing periods. The task-based usage patterns as well as their corresponding probabilities are depicted in the third and fourth column of the table. The upper part of the table shows that the user’s activity actually involves in multiple purposes. However, the user’s main intention is to perform online shopping as the probability of task #6 is significantly greater than the occurrence probabilities of other tasks, which means task #6 dominates the whole session. Therefore, we conclude that the dominant theme of the first user’s behaviour is actually more focused on task #6.

Table 6-5. Examples of task-oriented usage patterns

#	Real user session	Task # & title	Prob.
1	boutique search-result ProductDetailLegcare shopping_cart login2 Welcome expressCheckout your_account confirm_order vendor	Online shopping (#6) Product Legcare (#2) Boutique (#9) Department search (#1) Vendor info (3)	0.94 0.02 0.02 0.01 0.01
2	admissions/ admissions/requirements admissions/mailrequest admissions/orientation gradapp/appmain_right /news/default /programs/ programs/gradcs2002 programs/gradect2002 /programs/gradhci2002 /programs/core_guide	Admission (#4) Postgrad Program (#13)	0.63 0.37

For another user, we can find that the user is mainly conducting two tasks, i.e. task #4 and task #13. Referring to the derived tasks in Table 6-2, we can further identify that task #4 represents prospective students searching for admission information, such as requirement, orientation etc, whereas task #13 reflects the activity of those students who are particularly interested in postgraduate programs in IT disciplines. Unlike the first user, the second user clearly exhibits the cross-interest as the difference between the two corresponding probabilities of the tasks is not quite significant.

Once the task preference of a user is identified, it is possible to accurately recommend the preferred Web contents to the user.

6.4.5. Evaluation Result of Web Recommendation

From the viewpoint of users, the effectiveness of the proposed approach is evaluated by the precision of recommendation. Here, we exploit a metric called *hit precision* [29] to measure the effectiveness in the context of top-*N* recommendation. Given a user session in the test set, we extract the first *j* pages as an active session to generate a top-*N* recommendation set via the procedure described in section 6.3. Since the recommendation set is in a descending order, we then obtain the rank of $j+1$ page in the sorted recommendation list. Furthermore, for each rank $r > 0$, we sum the number of test data that exactly rank the *rth* as $Nb(r)$. Let $S(r) = \sum_{i=1}^{r} Nb(i)$, and $hitp = S(N)/|T|$, where $|T|$ represents the number of testing data in the whole test set. Thus, $hitp$ stands for the Web recommendation precision.

Obviously, the bigger the value of N (the number of recommendations) is the more accurate the recommendation is.

In order to compare our approach with other existing methods, we implement a baseline method that is based on a clustering technique [29]. This method is to generate usage-based session clusters by performing a *k*-means clustering process on usage data explicitly. Then the cluster centroids are derived as the aggregated access patterns.

Figure 6-1 depicts comparison results in terms of *hitp* parameter using the two methods discussed above on CTI dataset. The comparison results demonstrate the improvement of Web recommendation performance with the task-oriented usage mining in comparison with the traditional clustering-based algorithm in terms of recommendation precision. In this scenario, it can be concluded that our approach is capable of recommending user more preferable or interested Web content. In addition to the recommendation accuracy, this approach is able to identify the hidden tasks why such user sessions or Web pages are grouped together in a same category.

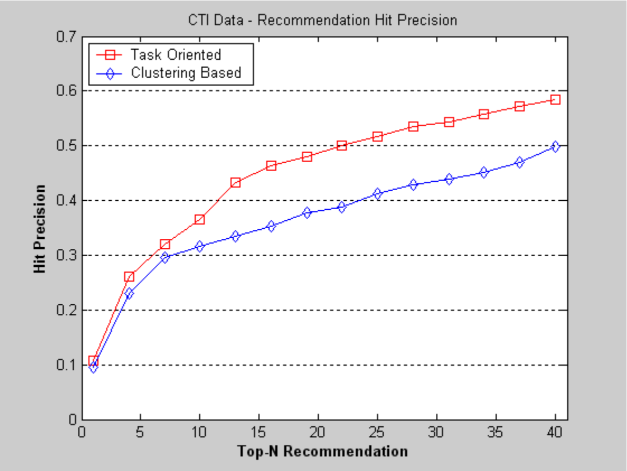


Figure 6-1. Web recommendation evaluation upon *hitp* comparison for CTI dataset

6.5. Conclusion

Web usage mining is a knowledge discovery process, which is not only to discover user access pattern, but also enable Web user to locate the needed information or content more efficiently via Web recommendations. In this chapter, we have developed a Web recommendation technique by exploiting the knowledge of usage pattern from Web usage mining based on PLSA model. With the proposed probabilistic method, we can measure the co-occurrence activities (i.e. user sessions) in terms of probability estimates to capture the underlying relationships among users and pages. Analysis of the estimated probabilities leads to building up the task-oriented usage patterns and Web page categories, and identifying the hidden factor space that conceptually representing user interests or tasks. The user navigational task distribution is computed via a Bayesian updating algorithm, in turn, is incorporated with the discovered factor space for making Web recommendations by using the top-N weighted scoring scheme. We have demonstrated the usability and effectiveness of our technique through experiments performed on the real world datasets.

7. User Profiling Algorithms for Web Recommendation Based on PLSA and LDA Model

7.1. Introduction

In chapter 6, we introduce a Web recommendation approach by identifying the user's task-oriented navigational distribution and incorporating it into the top-N weighted scoring scheme. Experiments conducted on the real world data sets have evaluated the proposed algorithm in terms of recommendation accuracy. The main idea of this approach is the use of the weights of pages within the dominant task space; however, it doesn't take the historical visits of other Web users into consideration. As a consequence, we aim to develop a Web recommendation algorithm via collaborative filtering techniques in this chapter. In particular, we propose two Web recommendation algorithms, which are called user profiling approaches based on two latent semantic analysis models.

The rest of this chapter is organized as follows: we first present two usage-based Web recommendation algorithms based on PLSA and LDA model respectively in section 7.2. In section 7.3, we give Web recommendation performance analysis in terms of recommendation accuracy from experiments conducted on the real world usage datasets, with the proposed recommendation algorithms described in section 7.2. In order to evaluate the effectiveness of the proposed algorithms, comparison studies are carried out against existing Web recommendation algorithms in section 7.4. Related work and conclusion are given in section 7.5 and 7.6, respectively.

7.2. User Profiling Algorithms for Web Recommendation

In this section, we present two usage-based user profiling algorithms for Web recommendation based on PLSA and LDA model respectively. In chapter 4 and 5, we have derived user access patterns and user profiles via a probability inference algorithm. In the following parts, we aim to incorporate the discovered usage knowledge with the collaborative filtering algorithm into the Web recommendation algorithm.

7.2.1. Recommendation Algorithm based on PLSA Model

As discussed in the previous chapter, Web usage mining will result in a set of user session clusters $SCL = \{SCL_1, SCL_2, \cdots SCL_k\}$, where each SCL_i is a collection of user sessions with similar access preferences. And from the discovered user session clusters, we can then generate their corresponding centroids of the user session clusters, which are considered as usage profiles, or user access patterns. The complete formulation of usage profiling algorithm is expressed as follows:

Given a user session cluster SCL_i, the corresponding usage profile of the cluster is represented as a sequence of page weights, which are dependent on the mean weights of all pages engaged in the cluster

$$up_i = \left(w_1^i, w_2^i, \cdots w_n^i\right) \tag{7.1}$$

where the contributed weight, w_j^i, of the page p_j within the user profile up_i is:

$$w_j^i = \frac{1}{|SCL_i|} \sum_{t \in SCL_i} a_{tj} \text{ ,} \tag{7.2}$$

And a_{tj} is the element weight of the page p_j in a user session $s_t, s_t \in SCL_i$. To further select the most significant pages for recommendation, we can use filtering method to choose a set of dominant pages with weights exceeding a certain value as an expression of user profile, that is, we preset a threshold μ and filter out those pages with weights greater than the threshold for constructing the user profile. Given w_j^i, then

$$w_j^i = \begin{cases} w_j^i, w_j^i > \mu \\ 0, otherwise \end{cases} \tag{7.3}$$

This process performs repeatedly on each user session cluster and finally generates a number of user profiles, which are expressed by the weighted sequences of pages. These usage patterns are then used into collaborative recommendation operations.

Generally, a Web recommendation is to predict and customize Web presentations in a user preferable style according to the interests exhibited by individual or groups of users. This goal is usually carried out in two ways. On the one hand, we can take the current active user’s historic behaviour or pattern into consideration, and predict the preferable information to this specific user. On the other hand, by finding the most similar access pattern to the current active user from the learned usage models of other users, we can recommend the tailored Web content. The former one is sometimes called memory-based approaches, whereas the latter one is called model-based recommendations, respectively. In this work, we adopt the model-based technique in our Web recommendation framework. We consider the usage-based user profiles generated in section 4.3 as the aggregated representations of common navigational behaviours exhibited by all individuals in the same particular user category, and utilize them as a usage knowledge base for recommending potentially visited Web pages to the current user.

Similar to the method proposed in [29] for representing user access interest in the form of a n-dimensional weighted page vector, we utilize the commonly used cosine function to measure the similarity between the current active user session and discovered usage patterns. We, then, choose the best suitable profile, which shares the highest similarity with the current session, as the matched pattern of current user. Finally, we generate the top-N recommendation pages based on the historically visited probabilities of pages by other users in the selected profile. The detailed procedure is described as follows:

[Algorithm 7.1]: User profiling algorithm for Web recommendation based on PLSA

[Input]: An active user session s_a and a set of user profiles $up = \{up_j\}$.

[Output]: Top-*N* recommendation pages $REC_{PLSA}(s_a) = \{ p_j^{mat} \,|\, p_j^{mat} \in P, j = 1, 2, \ldots, N\}$.

Step 1: The active session s_a and the discovered user profiles up are viewed as *n*-dimensional vectors over the page space within a site, i.e. $up_j = [w_1^j, w_2^j, \cdots, w_n^j]$, where w_i^j is the significant weight contributed by the page p_i in the up_j user profile, similarly $s_a = [w_1^a, w_2^a, \cdots w_n^a]$, where $w_i^a = 1$, if page p_i is already accessed, and otherwise $w_i^a = 0$,

Step 2: Measure the similarities between the active session and all derived usage profiles, and choose the maximal one out of the calculated similarities as the most matched pattern:

$$sim(s_a, up_j) = (s_a \cdot up_j) \big/ \|s_a\|_2 \, \|up_j\|_2 \tag{7.4}$$

where $s_a \cdot up_j = \sum_{i=1}^{n} w_i^j w_i^a$, $\|s_a\|_2 = \sqrt{\sum_{i=1}^{n} (w_i^a)^2}$, $\|up_j\|_2 = \sqrt{\sum_{i=1}^{n} (w_i^j)^2}$

$$sim(s_a, up_{mat}) = \max_j (sim(s_a, up_j)) \tag{7.5}$$

Step 3: Incorporate the selected profile up_{mat} with the active session s_a, then calculate the recommendation score $rs(p_i)$ for each page p_i:

$$rs(p_i) = \sqrt{w_i^{mat} \times sim(s_a, up_{mat})} \tag{7.6}$$

Thus, each page in the profile will be assigned a recommendation score between 0 and 1. Note that the recommendation score will be 0 if the page is already visited in the current session,

Step 4: Sort the calculated recommendation scores obtained in step 3 in a descending order, i.e. $rs = (w_1^{mat}, w_2^{mat}, \cdots, w_n^{mat})$, and select the N pages with the highest recommendation scores to construct the top-N recommendation set:

$$REC_{PLSA}(s_a) = \{ p_j^{mat} \mid rs(p_j^{mat}) > rs(p_{j+1}^{mat}), j = 1, 2, \cdots N-1 \} \tag{7.7}$$

7.2.2. Recommendation Algorithm Based on LDA Model

In this section, we present a user profiling algorithm for Web recommendation based on LDA generative model. As introduced in chapter 5, LDA is one of the generative models, which is to reveal the latent semantic correlation among the co-occurent activities via a generative procedure. Similar to the Web recommendation algorithm proposed in the previous section, we, first, discover the usage pattern by examining the posterior probability estimates derived via LDA model, then, measure the similarities between the active user session and the usage patterns to select the most matched user profile, and eventually make the collaborative recommendation by incorporating the usage patterns with collaborative filtering, i.e. referring to other users' visiting preferences, who have similar navigational behaviours. Likewise, we employ the top-N weighted scoring scheme algorithm into the collaborative recommendation process, to predict the user's potentially interested pages via referring to the page weight distribution in the closest access pattern. In the following part, we explain the details of the algorithm.

Given a set of user access models and the current active user session, the algorithm of generating the top-*N* most weighted pages recommendation is outlined as follows:

[Algorithm 7.2]: User profiling for Web recommendation based on LDA model

[Input]: An active user session s_a, the computed session-topic preference distribution θ and a predefined threshold μ.

[Output]: Top-*N* recommendation pages $REC_{lda}(s_a) = \{p_j^{rec} \mid p_j^{rec} \in P, j = 1, \cdots N-1\}$.

Step 1: Treat the active session s_a as a n-dimensional vectors: $s_a = [w_1^a, w_2^a, \cdots, w_n^a]$, where $w_i^a = 1$, if page p_i is already clicked, and otherwise $w_i^a = 0$,

Step 2: For each latent topic z_j, choose all user sessions s_i with $\theta_{z_j}^{s_i} \geq \mu$ to construct a user session aggregation *R* and compute the usage pattern as

$$up_j = \frac{\sum\limits_{s_i \in R} \theta_{z_j}^{s_i} \cdot s_i}{|R|} \tag{7.8}$$

Step 3: Measure the similarities between the active session and all learned user access models, and choose the maximum one out of the calculated similarities as the most closely matched access pattern:

$$sim(s_a, up_j) = (s_a \cdot up_j) \big/ \|s_a\|_2 \, \|up_j\|_2 \tag{7.9}$$

, where $s_a \cdot up_j = \sum_{i=1}^{n} w_i^a w_i^j$,$\|s_a\|_2 = \sqrt{\sum_{i=1}^{n} (w_i^a)^2}$, $\|up_j\|_2 = \sqrt{\sum_{i=1}^{n} (w_i^j)^2}$

$$sim(s_a, up_{rec}) = \max_j(sim(s_a, up_j)) \tag{7.10}$$

Step 4: Refer to the page weight distribution in the most matched access pattern up_{rec}, and calculate the recommendation score $RS(p_i)$ for each page p_i:

$$RS(p_i) = \sqrt{w_i^{rec} \times sim(s_a, up_{rec})} \tag{7.11}$$

Thus, each page in the matched pattern will be assigned a recommendation score between 0 and 1. Note that the recommendation score will be 0 if the page is already visited in the current session,

Step 5: Sort the calculated recommendation scores obtained in step 4 in a descending order, i.e. $RS = (w_1^{rec}, w_2^{rec}, \cdots, w_n^{rec})$, and select the N pages with the top-N highest recommendation scores to construct the top-N recommendation set:

$$REC_{LDA}(s_a) = \{p_j^{rec} \mid RS(p_j^{rec}) > RS(p_{j+1}^{rec}), j = 1, 2, \cdots N-1\} \tag{7.12}$$

7.3. Experiments and Results

In order to evaluate the effectiveness of the proposed methods based on PLSA and LDA model, we conduct experiments on two real world usage data sets, and conduct comparisons with existing recommendation algorithms. We present the results of the recommendation performance in this section.

7.3.1 Data Sets

The Web usage dataset used in experiments is the same as those used in the previous chapters. We briefly describe the datasets as follows. The data set is from a university's department Web log file, which consists of 13745 sessions and 683 pages, and each entry of the usage matrix corresponds to the amount of time (in seconds) spent on pages during a given session. For convenience, we refer to this data as “CTI data”.

7.3.2. Evaluation Metric of Web Recommendation Accuracy

Here we use the evaluation metric of Web recommendation accuracy described in the previous chapter. This metric is called *hit precision* [29], which is used to assess the effectiveness of the recommendation algorithm in the context of top-*N* recommendation.

In order to compare our approach with other existing methods, we implement a baseline method that is based on the clustering technique [29].

7.3.3. Experimental Results

Figure 7-1 depicts the comparison results of recommendation accuracy in terms of *hitp* parameter using PLSA-based and clustering-based recommendation algorithm respectively with CTI dataset. From the Figure 7-1, it is shown that the proposed PLSA-based technique consistently overweighs the standard clustering-based algorithm in terms of hit precision parameter. In this scenario, it can be concluded that our approach is capable of making Web recommendation more accurately and effectively against the conventional methods.

For another user, we can find that the user is mainly conducting two tasks, i.e. task #4 and task #13. Referring to the derived tasks in Table 6-2, we can further identify that task #4 represents prospective students searching for admission information, such as requirement, orientation etc, whereas task #13 reflects the activity of those students who are particularly interested in postgraduate programs in IT disciplines. Unlike the first user, the second user clearly exhibits the cross-interest as the difference between the two corresponding probabilities of the tasks is not quite significant.

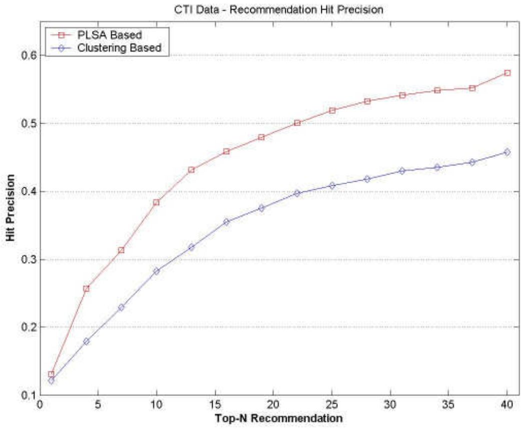


Figure 7-1. Web recommendation evaluation upon *hitp* comparison for CTI dataset

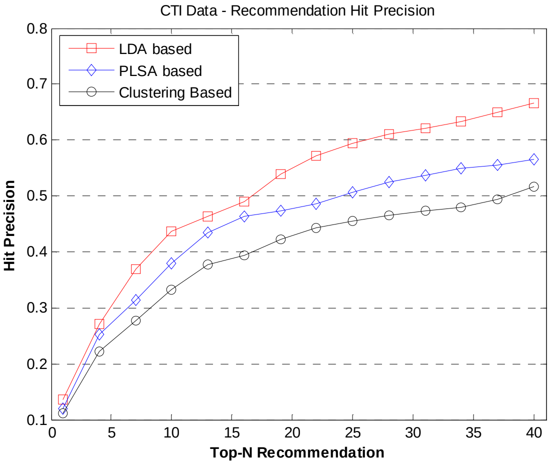


Figure 7-2. Hit precision comparison of Web recommendation on CTI dataset

The experimental results in terms of hit precision with LDA model are shown in Figure 7-2. In order to compare the proposed approach with other methods, we also carry out experiments on CTI dataset with the conventional clustering-based and the PLSA-based approaches. In a similar manner, the usage-based session clusters by performing the k-means clustering and the probability inference with PLSA model [29, 96] are constructed to aggregate user sessions with similar access preferences, and the centroids of clusters are derived as the aggregated user access patterns.

The results demonstrate that the proposed LDA-based technique consistently outperforms the standard clustering-based and the PLSA-based algorithms in terms of hit precision parameter, the standard clustering-based algorithm always generates the least accurate recommendation precision, and the recommendation performance of the PLSA-based algorithm is in the middle of the other two. From this comparison, it can be concluded that the proposed recommendation approaches based on latent semantic analysis models are capable of making Web recommendation more accurate and effective against the conventional recommendation methods. In addition to the advantage of high recommendation accuracy, these approaches are also able to identify the latent semantic factors why such user sessions or Web pages are grouped together in the same category.

7.4. Related work

Web recommendation research has become a hot topic in the context of Web data management in the last decade despite of the fact that recommender systems have been well studied in machine learning and information retrieval areas.

To-date, there are two kinds of approaches and techniques commonly used in Web recommendation, namely content-based filtering and collaborative filtering systems [38, 39]. Content-based filtering systems such as WebWatcher [40] and client-side agent Letizia [41] generally generate recommendations based on the pre-constructed user profiles by measuring the similarity of Web contents to these profiles, while collaborative filtering systems make recommendations by utilizing the rating of the current user for objects via referring to other users' preferences that is closely similar to the current one.

In addition, Web usage mining has been proposed as an alternative method for not only revealing user access patterns, but also making Web recommendations recently [29]. With the benefit of great progress in data mining research communities, many data mining techniques, such as collaborative filtering based on the k-Nearest Neighbour algorithm (*kNN*) [42-44], Web user or page clustering [29, 45, 46], association rule mining [47, 48] and sequential pattern mining technique [19] have been adopted in the current Web usage mining methods. With the development of Web usage mining techniques contributed by academics and researchers in a variety of application areas, the application fields are broadened widely and deepened throughout. For example, Liu et al [97] proposed a framework for forming communities in a peer-to-peer communication environment by analysing the client-side Web browsing history. This framework is based on a statistics-based approach. In [98], Bose et al incorporated the ontology in the form of concept hierarchy into usage based recommendation systems to reinforce recommendations by taking semantics into account, whereas [99] combined model-based and memory-based CF algorithms into a hybrid system to improve the recommendation performance without incurring high computational costs. And Jin et al [100] introduced a Maximum Entropy algorithm into the recommendation scoring algorithm to achieve better recommendation. Consequently, many efforts have been contributed and great achievements have been made in such research fields as Web personalization and recommendation systems [40, 41, 49, 50], Web system improvement [51], Web site modification or redesign [46, 52], and business intelligence and e-commerce [5].

7.5. Conclusion

Web transaction data between Web visitors and Web pages usually convey user task-oriented behaviour patterns. As a result, there is an increasing demand to develop techniques that can not only discover user task-oriented access patterns, but also characterize the underlying relationships among Web users, user access tasks and Web pages.

In this chapter, we have proposed a unified user profiling algorithm for Web recommendation based on PLSA and LDA model. With the discovered usage knowledge from Web usage mining via various latent semantic analysis models, we construct a set of usage access patterns (i.e. user profiles). By measuring the similarities between the active user and the discovered usage patterns, we choose the most similar user profile as the candidate usage pattern. The recommended page list is generated by incorporating the chosen user profile with the top-N weighted scoring scheme for Web recommendation.

We have developed two Web recommendation algorithms with PLSA and LDA model respectively. Experimental results that conducted in comparison with other existing recommendation algorithms have shown that latent semantic analysis based recommendation algorithms are able to make recommendations accurately and efficiently. In addition to the high recommendation precision, the latent semantic analysis models have the capability of revealing the latent factor space associated with the discovered usage knowledge.

8. Case Studies of Clustering-Based User Profiling Techniques in Gait Pattern Mining

8.1. Introduction

In the previous chapters, we intensively discuss Web usage mining for discovering user access patterns, and for predicting user navigational preferences and recommending the customized Web contents to Web users. During this procedure, clustering-based user profiling techniques (CBUP) plays an important role for usage knowledge discovery due to the capability of capturing the latent aggregate nature of co-occurrence observations. In addition to its application in the area of Web information processing, CBUF can also be applied into a wide range of knowledge discovery and management domains, for example, in biomedical or health knowledge discovery and management fields, CBUF is usually used to create various typical characteristics to represent specific patient groups, which can be considered as pathological indicatives for various types of disorders or disease symptoms.

In this chapter, we aim to extend our developed methodologies and algorithms of CBUF from Web usage mining to a healthcare-based application, i.e. gait analysis, to investigate the hidden correlation among gait variables, discover normal and abnormal gait patterns in the form of gait variable vector, and eventually explore the applicability of gait pattern mining in the diagnosis and analysis of human movement capability disorder. We carry out two case studies of gait pattern mining and give experimental results in this chapter.

This chapter is organized as follows. In section 8.2, we present traditional clustering based algorithms for a case study of *Cerebral Palsy* (CP) patients. CP gait variable model and clustering algorithms are discussed in this section. Then, we perform another case study on monitoring the changes of gait characteristics in an elder population, which are considered in relation to fall risks in section 8.3. In particular, we employ a SOM-based clustering algorithm to reveal gait patterns. Experimental analysis on two gait datasets is carried out to assess the proposed techniques. Related work is discussed in section 8.4. And we conclude this chapter in section 8.5.

8.2. Case Study of Gait Pattern Mining in CP Patients

In this study, we aim to investigate gait pattern analysis of CP patients using a gait data model of temporal-distance gait variables. Traditional clustering algorithms are employed to address gait pattern mining.

8.2.1. Gait Data Model in Gait Pattern Mining

As for a biomechanical application of gait analysis, there are a variety of basic temporal-distance parameters that are frequently used for modelling human walking, such as walking speed, stance/swing times. This may be due to the fact that the temporal-distance parameters are probably more fundamental for the purpose of gait analysis [101]. In this work, we simply exploit the specific two-dimensional temporal-distance parameters, i.e. stride length and step frequency/cadence to construct a gait data model. Both normal and pathological data relating to children’s gait information for developing gait models were taken from [102]. In this model, the gait data is expressed as a two-dimensional feature vector matrix, in which each row represents a subject vector in terms of stride length and cadence parameters, whereas each column is corresponding to the selected gait variable. In the following experiments, we will conduct data analysis on the constructed gait data to reveal the individual-specific gait patterns. In addition to kinematic parameters, other two physical features, i.e. leg length and age, are taken into consideration for normalizing and scaling to eliminate the impact of the diversity in individuals.

Normalization and Scaling

To remove the relative difference within the gathered gait data in terms of subject's age and leg length and leave the pathological trends, a polynomial-based normalization technique [102] is employed on the stride length and cadence parameters respectively:

$$NSL = SL - (a_0 + a_1 LL + a_2 (LL)^2 + \cdots + a_k (LL)^k) + \overline{SL}_N \tag{8.1}$$

$$NCAD = CAD - (b_0 + b_1 AGE + b_2 (AGE)^2 + \cdots + b_k (AGE)^k) + \overline{CAD}_N \tag{8.2}$$

where *NSL,SL LL*, *NCAD*, *CAD*, *AGE* are subject's (normal or pathological) normalized stride length, original stride length, leg length, normalized cadence, original cadence and actual age respectively, $\overline{SL}_N$ and $\overline{CAD}_N$ stand for average stride length of intact subjects and average cadence of intact subjects.

Since Euclidean distance is employed to measure the similarity between two subjects' gait characteristics, a scaling process on gait data is needed to have unity variance and decrease the influence of one feature dominating the distance over another feature with its significant value.

$$SNSL = C_{SL} NSL \tag{8.3}$$

$$SNCAD = C_{CAD} NCAD \tag{8.4}$$

where *SNSL* and *SNCAD* are subject's stride length and cadence after normalizing and scaling, C_{SL} and C_{CAD} are coefficients for stride length and cadence scaling.

Similarity Measurement

After data normalization and scaling transformation, the discrepancy not only in individual's physical condition (i.e. leg length and age), but also in the observation variance of stride length and cadence caused by one feature dominating another in value, will be removed. Then, one basic similarity metric, i.e. Euclidean distance that is well-adopted to measure the distance of two feature vectors in Information Retrieval [61], is utilized to measure the similarity of two subjects since every gait data could be considered as a feature vector in this case.

$$sim(s_i, s_j) = d_2(s_i, s_j) = \sqrt{\sum_{t=1}^{2}(g_{it} - g_{jt})^2} \tag{8.5}$$

where s_i is the *i*-th subject of gait data, g_{it} denotes the chosen kinematic value of s_i on *j*-th variable.

Moreover, since the centroid of the subject cluster could be virtually viewed as a subject in the form of feature vector, the distance between the generated centroid and the individual subject could be further expressed as the affiliated distance of this subject from the subject group.

$$AD(s_i, C_k) = d_2(s_i, cid_k) \tag{8.6}$$

In clustering stages, this kind of distance is calculated repeatedly until the mean distance converges to a local optimal value.

8.2.2. Clustering-based User Profiling Algorithms for Gait Pattern Mining

Two types of clustering algorithms, i.e. *k*-means and hierarchical clustering are conducted to group gait data in terms of temporal-distance parameters. In *k-means* clustering analysis, we investigate the implementation of grouping the ambulation of neurologically intact individuals and those with CP into *k* subject categories, visualizing the separation layout of the grouped subject cluster and evaluating the clustering quality in terms of mean silhouette and mean square error. The *k-means* clustering algorithm works as follows [53-55]:

[Algorithm 8.1]: *k*-means clustering for gait pattern mining

[Input]: Subject gait data matrix in the form of temporal-distance variables.

[Output]: A set of subject gait clusters and corresponding centroids.

Step 1: Arbitrarily choose *k* subjects as initial cluster mean centres;

Step 2: Then assign each subject to the cluster with the nearest centres, and update each mean centre of the cluster;

Step 3: Repeat step 2 until all centres don't change and no reassignment is needed;

Step 4: Finally output subject clusters and their corresponding centroids.

In addition to *k*-means clustering, hierarchical clustering is also employed to reveal the possible grouping strategy for gait data from the viewpoint of hierarchy tree analysis. Meanwhile, construction of hierarchy tree and its corresponding visualization layout of clusters as well as centroids are plotted for comparing the clustering results derived by these two kinds of clustering algorithms. The procedure of hierarchy clustering is as below [17]:

[Algorithm 8.2]: Hierarchical clustering for gait pattern mining

[Input]: A set of subject gait data in the form of temporal-distance variables.

[Output]: A hierarchy cluster tree and its corresponding visualization.

Step 1: Calculate the mutual distance of paired subjects (distance matrix) as the clustering criteria;

Step 2: Decompose subject dataset into a set of levels of nested aggregation based on the distance matrix (i.e. tree of clusters);

Step 3: Cut the hierarchical tree at the desired levels by selecting a predefined threshold, and then explicitly merge all connected subjects below the cut level to create various clusters;

Step 4: Output the dendrogram and cluster visualization.

8.2.3. Experiments and Results of CP Gait Analysis

Experimental Data and Design

The stride length and cadence gait data of 68 normal children and 88 children with CP are constructed as a temporal-distance gait data from [102]. In order to accomplish normalization process described above, we utilize a first-order and a second-order polynomial models for normalizing stride length and cadence respectively. The engaged coefficients are tabulated in Table 8-1 [102]. In addition, the scaling factors that are used to unify the amplitude in variance of stride length and cadence parameters are listed in Table 8-1 as well. Figure 8-1 illustrates the normalized two-dimensional plot of gait data, i.e. stride length vs. cadence, for 68 neurological intact children and 88 children with CP. In this figure, red solid dots stand for the subjects in neurological intact group, whereas black cross symbol represents the subject with pathological symptoms. Consequently, our aim is to separate these two main types of subjects into various groups, within which the subjects should share the similar gait characteristics. Especially, after clustering, the subjects in neurological intact group should be ideally categorized into the same cluster.

Table 8-1. Normalization coefficients and scaling factors for gait data

name	a_0	a_1	$\overline{SL}$	b_0	b_1	b_2	$\overline{CAD}$	c_{SL}	c_{CAD}
value	0.28	1.31	1.02	174.07	-7.04	0.22	136.84	5.88	0.034

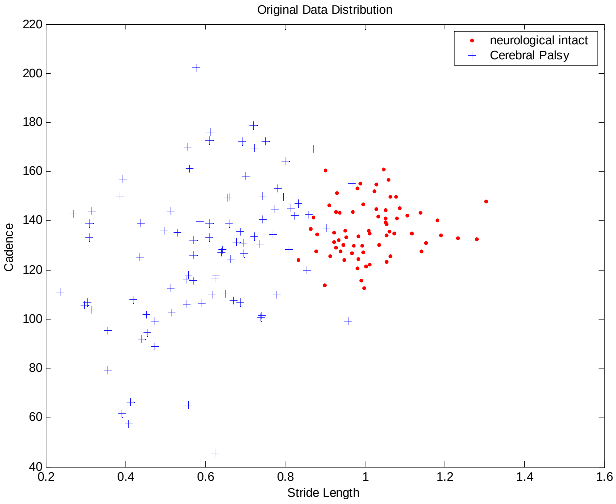


Figure 8-1**.** Normalized gait data for children in normal and pathological groups

Experimental Results

The clustering results with respect to *k*-means are visualized in Figure 8-2 (a)-(b) for $k = 5, 6$ respectively, where the grouped subjects are symbolized with a variety of point types and colours. In addition, the corresponding centroids of clusters are marked in black solid dots in the figures as well. From these plots, it is visually demonstrated which subjects are grouped together into the same cluster according to their mutual Euclidean distance, how close the subjects within the same cluster are and how far the subjects are separate from others in different clusters. For example, the neurologically intact subjects are almost partitioned into the first cluster with blue square in case of $k = 5$, while for

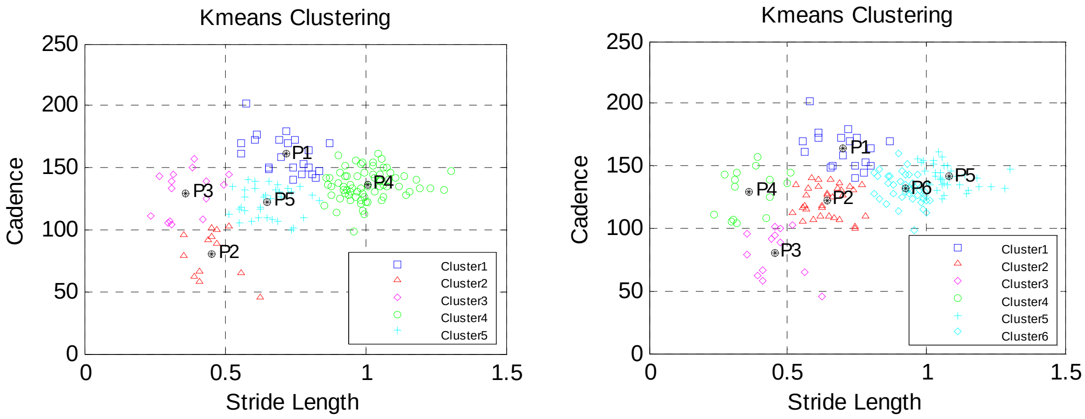
$k = 6$, such subjects are separated into two individual clusters, in which they are represented by blue squares and cyan cross symbols accordingly.

Table 8-2. Centroids of clusters with k-means when $k = 5$

Centroid #	Stride Length	Cadence
P1	0.7190	160.8955
P2	0.4557	80.6862
P3	0.3630	129.6800
P4	1.0102	136.0906
P5	0.6533	122.2424

Table 8-3. Centroids of clusters with k-means when $k = 6$

Centroid #	Stride Length	Cadence
P1	0.7024	163.4505
P2	0.6428	121.9912
P3	0.4557	80.6862
P4	0.3630	129.6800
P5	1.0811	141.4067
P6	0.9552	131.5741



(a) (b)

Figure 8-2**.** Cluster visualization of *k*-means with $k = 5$ (a) and $k = 6$ (b)

Meanwhile, the centroids of clusters for $k = 5$ and $k = 6$ are tabulated in Table 8-2 and Table 8-3 respectively, which are indicated by the point sequence number ranging from 1 to 6 in the cluster layout (i.e. Figure 8-2 (a)-(b)). Furthermore, the created centroids can be treated as the pattern-based gait profiles to represent the overall gait characteristics of the corresponding gait groups [103].

In order to evaluate the quality of clustering, we introduce two basic coefficients, namely *Silhouette Coefficient* (SC) and *Mean Square Error* (MSE) in this study.

Firstly, we compare clustering quality with respect to a variety of parameter settings of cluster numbers (i.e. *k* value). In order to be independent from the number of clusters produced, we use the silhouette coefficient for the purpose of evaluation.

The silhouette coefficient *SC* is an indicator to measure the quality of clustering, which is normally a value between 0 and 1, and rather independent from the number of clustering. Theoretically, the larger the value of *SC* is, the higher the quality of the cluster will be.

In addition to silhouette coefficient, we also conducted further evaluation study on overall mean errors of clustering rather than on one single cluster quality, for the purpose of comparison [104].

It is easily concluded that the *MSE* stands for the overall mean distance for each subject within the same cluster from its corresponding centroid, which reveals the quality of clustering as well.

Figure 8-4 summarizes the calculated results in terms of SC and *MSE* for *k*-means clustering in case of $k = 4,5,6$. Interestingly, the table shows that the highest value for SC is for $k = 5$, $k = 4$ and $k = 6$ rank the second and third, whereas the smallest mean square error occurs in $k = 6$ instead of $k = 5$. This is mainly because one neurological intact group is split into two individual sub-clusters, which will result in the decrease of distance from every subject in the sub-cluster to its centroid of the sub-cluster.

Table 8-4. Mean silhouette and mean square error for k-means with $k = 4, 5, 6$

K	4	5	6
SC	0.5981	0.6408	0.5510
MSE	134.95	97.42	72.26

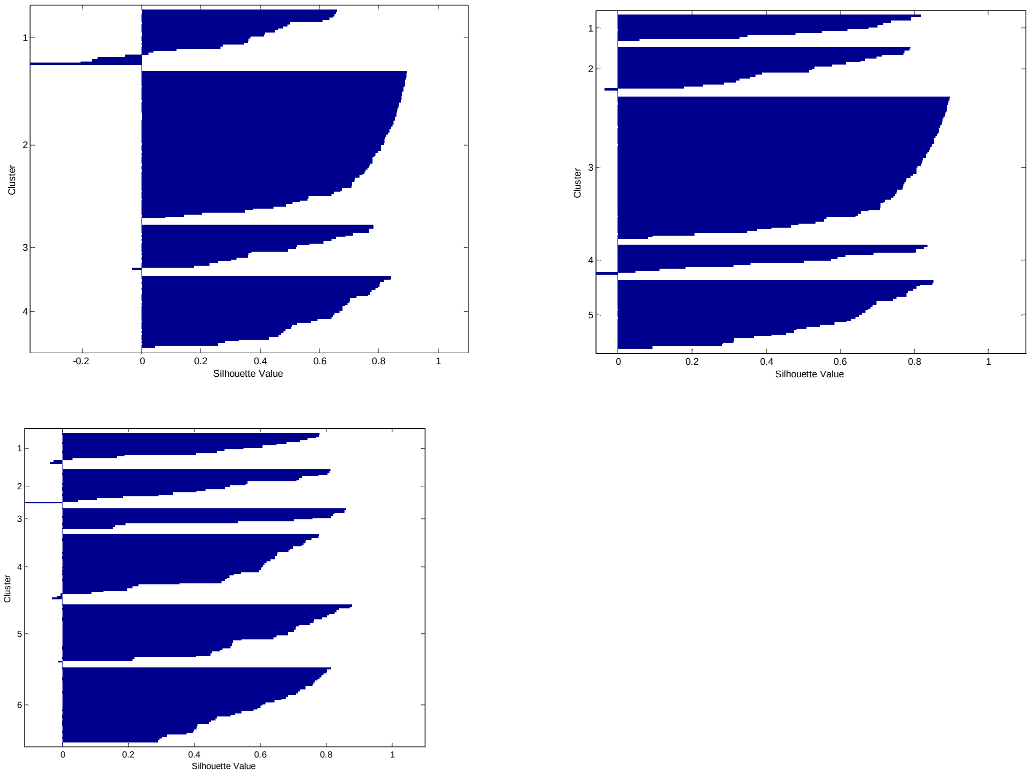


Figure 8-3**.** Silhouette value plots of *k*-means clustering with $k = 4, 5, 6$

Figure 8-3 shows the silhouette plots of *k*-means clustering for various cluster number settings. From this plot, it is shown that there is one particular cluster that seems to be rather well-separated, which actually consists of neurological intact subjects, while others are not distinct enough for the three different k settings. In addition, the plots indicate that the clusters generated with $k = 5$ exhibit a little bit higher quality than other two k settings, which is also validated by *SC* shown in Table 8-4. Especially, the negative values of *SC* reflect that the corresponding subjects are partitioned wrongly into inappropriate subject groups, according to its definition. Consequently, the bigger occurrence rate of negative SC reveals the poor quality of clustering accordingly. From this viewpoint, it is concluded that the selection of cluster number with $k = 5$ is much more appropriate than those of cluster number settings with $k = 4, 6$. Conclusively, we will stick to selection of $k = 5$ to conduct hierarchical clustering and test data validation in the following analysis.

Hierarchical Clustering

In comparison with *k*-means clustering, we also investigate the partition of gait data via a hierarchy tree approach. Hierarchy clustering is to create a hierarchical tree of clusters based on the mutual distance between each pair of observations.

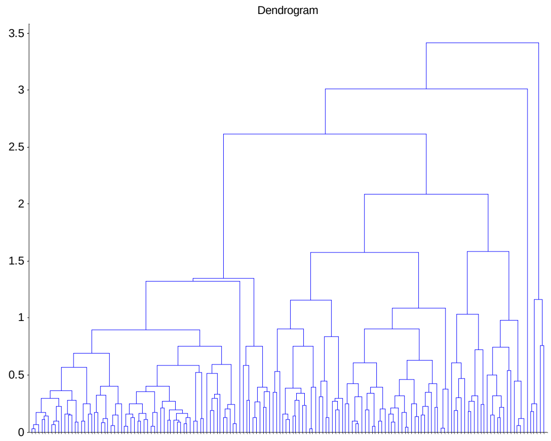


Figure 8-4**.** Hierarchical cluster tree

Figure 8-4 and Figure 8-5 illustrate the plotted hierarchical tree of clusters in the form of Dendrogram and visualized cluster layout of hierarchy tree respectively. In Dendrogram, the *x* coordinate stands for the processed subject sequence number, whereas y coordinate conveys the distance information between two adjacent nodes. Interestingly, the first 68 *x* coordinates in the dendrogram are exactly the same as the subject orders in the original gait dataset, which will result in the production of the first big subject cluster. However, the figure indicates further that there exist several oddish subjects, which are far enough from other objects so that they could not be grouped into either of the two established clusters in the lower level.

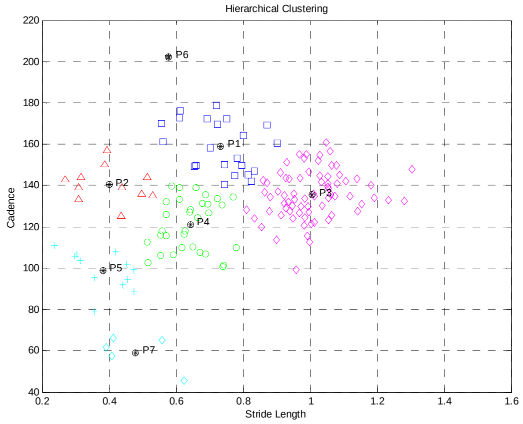


Figure 8-5. Cluster visualization of hierarchy tree

Table 8-5. Centroids of clusters with hierarchy clustering

Centroid #	Stride Length	Cadence
P1	0.7338	158.99
P2	0.4002	140.48
P3	1.0065	135.49
P4	0.6424	120.97
P5	0.3821	98.82
P6	0.5781	202.38
P7	0.4789	59.14

8.3. Case Study of Gait Pattern Mining in Ageing Using SOM-Based Clustering

8.3.1. Gait Data Model and SOM-based Clustering Algorithm

Gait Feature

In the first case study, we simply exploit the specific two-dimensional temporal-distance parameters, i.e. stride length and step frequency/cadence, to construct gait data model, and gait data is expressed as a two-dimensional feature vector matrix, in which each row represents a subject vector in terms of stride length and cadence parameters, whereas every column is corresponding to a gait variable.

In this section, we adopt another complicated type of gait variables, i.e. *Minimum foot clearance* (MFC) to model human walking characteristics. MFC occurring in the mid-swing phase of the gait cycle, is defined as the minimum vertical distance between the lowest point under the front part of the shoe/foot and the ground, and has been identified as an important gait parameter [105]. As a result, this parameter measures fall-prone gait characteristics and provides valuable information for identification of fall-risk due to degeneration of mobility in elderly population. Figure 8-6 illustrates what MFC parameter stands for in one gait cycle.

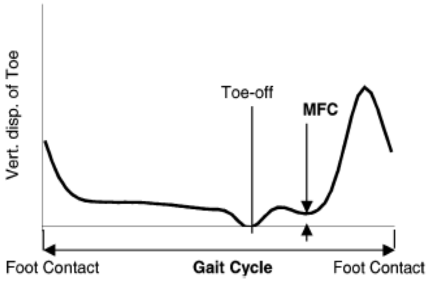


Figure 8-6. Vertical displacement of toe marker for one gait cycle during walking

8.3.2. SOM-based Clustering Algorithm

Self Organization Map (SOM) is a neural network based learning algorithm, which is to assemble input subject data together based on the mutual distance. In a SOM learning process, the subjects with similar patterns are aggregated together in closely neighbouring parts of an appropriately defined grid. In the context of gait analysis, SOM algorithm can be used as an analytical tool, which can efficiently visualize the gait pattern distribution in a two-dimensional SOM grid by reducing the dimensionality of input data with minimal loss of information content and no *priori* definition of clusters.

The SOM process consists of a regular, usually two-dimensional, grid of map units. Each unit *i* is represented by an n-dimensional prototype vector, $m_i = [m_{i1}, \cdots, m_{in}]$ where *n* is the dimension of input space. In the grid, the units are connected to adjacent ones by a neighbourhood relation. The number of map units, which varies depending on the size of input space, determines the accuracy and generalization capability of the SOM. Thus, the SOM can be considered as a topology preserving mapping from input space onto the two-dimensional grid of map units [106].

At each learning step, a data sample *x* is selected and the nearest map unit (i.e. *best matching unit*, BMU) is found on the map. The prototype vector of the BMU and its neighbouring units on the grid are merged toward the sample vector:

$$m_i(t+1) = m_i(t) + \alpha(t) h_{bi}(t)[x - m_i(t)] \tag{8.7}$$

where $\alpha(t)$ is a learning rate and $h_{bi}(t)$ is a neighbourhood kernel centred on the winner unit. Both learning unit and neighbourhood kernel radius decrease monotonically with time.

The SOM operation is trained iteratively where input vectors are aggregated until the following error function reaches the minimum:

$$E = \sum_{i=1}^{N} \sum_{j=1}^{M} h_{bj} \left\| x_i - m_j \right\|^2 \tag{8.8}$$

where *N* is the number of training data and *M* is the number of map units.

As SOM is to graphically map the input data space onto a trained grid where the data with similar characteristics are assembled onto neighbouring units of grid, therefore, it is possible to conduct clustering on the input samples via examining the distribution of the trained grid units. The most commonly used methods for visualizing the cluster structure of SOM are based on distance matrix, especially the unified distance matrix (U-matrix), which exhibits the distance between prototype vectors of neighbouring map units. In the experiment part, we will demonstrate the process of discovering gait pattern by using U-matrix. As a result of capability of handling data analysis with high dimensionality, SOM-based analysis has recently become a feasible visualization tool in pattern mining [107, 108].

8.3.3. Experiments and Results of Gait Analysis in Elder Population

Gait Dataset

The MFC dataset for experiments includes 78 subjects in three gait categories, i.e. 30 younger subjects, 38 elderly subjects with healthy gait and 10 elderly subjects with impaired walking ability are taken from the gait database of the Biomechanics Unit of Victoria University. Obviously, it is supposed that there are some gait overlaps in these three groups due to the diversity of individual physical condition. Especially, it occurs frequently in the second participant group. Our aim is to not only differentiate these three groups, but also discover the possibly existing gait patterns between them.

While the well-used MFC histogram plots show valuable statistical characteristics of the distribution in biomechanics, MFC plots of successive gait cycles between MFC_n and MFC_{n+1} illustrate unique interaction effects in 2-D gait analysis. Such plots, known as Poincaré Plots, have been shown to be highly effective in studying repetitive events. Features characterizing these plots are used to develop gait data model. Table 8-6 illustrates 9 relevant features dominating in gait analysis as well as their mathematical definitions extracted from Poincaré Plots, which is constructed as another gait variable space for gait pattern mining. All the gait features are normalized using their z-scores to have zero mean and unity variance for eliminating the individual physical diversity.

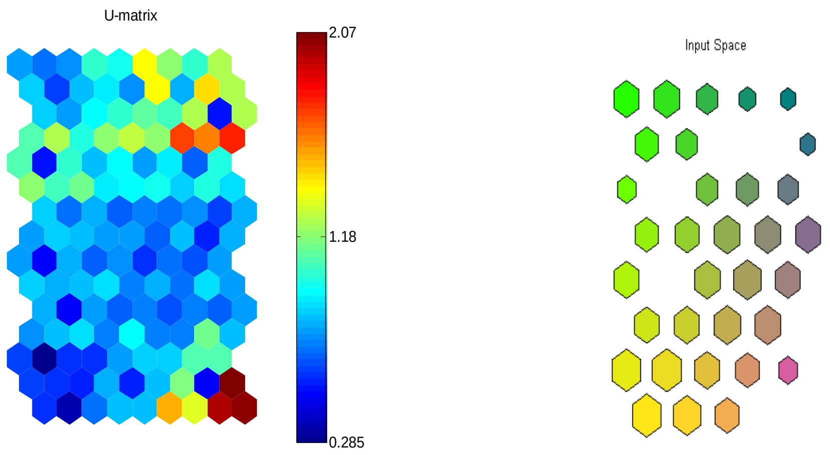
Table 8-6. Features and corresponding definitions extracted from MFC Poincaré Plots

Attribute #	definition	Attribute #	definition
1	$MFC_{\min}$	6	Mean
2	$MFC_{\max}$	7	Standard Deviation
3	$1^{st}\,Quartile(Q1)$	8	Skewness
4	$2^{nd}\,Quartile/Median(Q2)$	9	Kurtosis
5	$3^{rd}\,Quartile(Q3)$		

In the second experiment, we aim to employ SOM-based clustering techniques to discover the potentially existing gait characteristic patterns from gait data of these three human groups, which might possibly reflect different linking to fall risks. As discussed above, a visual inspection on U-matrix, which is derived from SOM training, is performed to visually find the gait group. Then, *k*-means (or called partitive) clustering was applied to further identify the distinct areas separated in the trained SOM map.

Visualization Inspection.

In Figure 8-7, two SOM visualization figures of MFC dataset are shown: the U-matrix (a) and the trained map units in the input space (b), which are illustrated in various colour representations. In Figure 8-7(a) of U-matrix, different colours reflect the mutual distance between map units, whose values are determined by the colour bar displayed beside the U-matrix map, e.g. the upper of colour bar is, the bigger value of distance is. As a result of colour representation of U-matrix, it is possible to visually identify the agglomerative areas/clusters in the trained map. From Figure 8-7(a), one can detect there seem two distinct clusters existing at the upper part of the SOM map. However, at the lower part of the SOM map, there may exist many sub-agglomerative blocks mixed with each other, which indicates the presence of overlaps of gait groups. In addition, there exist two small clusters corresponding to outliers, which are located at two corners of right side of the SOM map. Examining the trained map units in the input space (Figure 8-7(b)) would draw similar conclusion to that from U-matrix. Note that the size of each unit indicates the number of hits in each unit of input space, in particular, the empty units represent the zero-hits unit such that could be considered as interpolative units. And the interpolative units seem to divide the gait data into upper and lower two major parts, where there seem no clearly distinct sub-classes in lower part, indicating the presence of overlapping of gait patterns.



(a) (b)

Figure 8-7**.** The U-matrix (a) and prototype vectors of the SOM in input space (b)

Meanwhile, *Sammon* mapping technique is also performed to make a comparison of visual inspection, which is shown in Figure 8-8. From the Figure, one can similarly find that these three gait groups are not well separated from each other, especially being heavily overlapped at left side of the grid map.

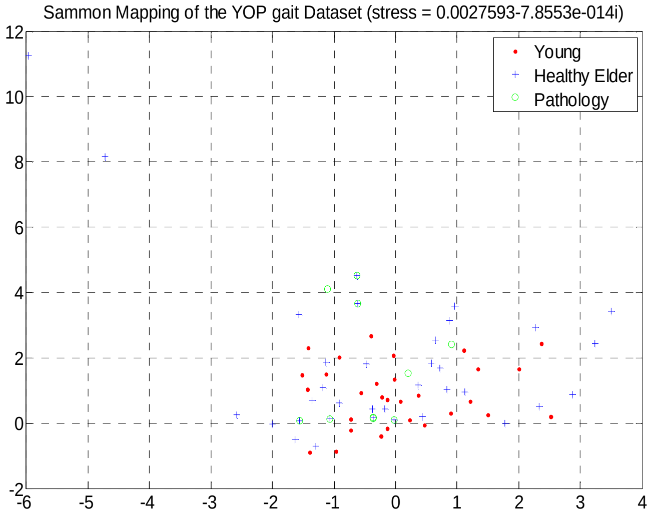


Figure 8-8. Sammon mapping grid of MFC dataset

Partitive Clustering

To further quantitatively analyse the underlying gait pattern hidden in MFC dataset, we performed partitive clustering i.e. *k*-means on the trained SOM grid to partition the map. Figure 8-9 depicts the clustering results. In *k*-means clustering algorithms, initial number of clusters is needed to set in advance, and different number setting will result in various cluster results. To select a best number of clusters, here a metric, called B-E index is used to measure clustering quality. That is, the smaller the B-E index is, the better the quality of clustering is. From the figure, it is shown that the plot of B-E index has a negative peak at seven clusters. Furthermore, seven clusters, which are represented by seven colours, are illustrated on the SOM map, based on the best selected cluster number. From the calculated clusters, the agglomerative properties of these three gait categories are clearly visualized, in turn, are used to model the potentially existing gait patterns.

Pattern-Specific Gait Locality Analysis using Histogram Plot

Once a number of gait clusters are visually discovered, it is necessary to further investigate the linking between each gait clusters on the SOM and clinic-specific gait patterns. For the three gait categories, i.e. Younger, Elder and Healthy, Elder with impaired balance, we aim to perform histogram analysis on the SOM grid to determine the locality of clinic-specific gait pattern on the SOM grid. Histogram plots can quantitatively illustrate the gait distribution over the trained map based on the hit numbers of input space, which is counted by considering the minimum distance between each input data and trained prototype vectors as a hit. Figure 8-10 depicts the histograms of three categories of input gait data on the SOM. Three gait categories are represented by red, green and blue colour hexagons, whose sizes reflect the hits volumes. In the Figure, it is shown that the younger group’s gaits are mainly distributed over two areas of SOM map: left upper side and lower middle part, elder and healthy population’s gaits are scattered almost all map, and Elder with impaired balance group exhibits walking characteristics commonly located at the left lower corner and middle part of the SOM. Furthermore, we employ fuzzy histogram analysis to investigate these three gait groups’ gait distribution over the SOM, which is to calculate the hits volume by accumulating each input data’s affiliated memberships to all trained map units instead of strictly selecting the most closed unit with minimum mutual distance. The fuzzy histograms are also displayed in Figure 8-10 in comparison to standard histograms. The fuzzy histograms further justify the conclusion of the pattern-specific gait locality distribution over the SOM map derived via the clustering analysis.

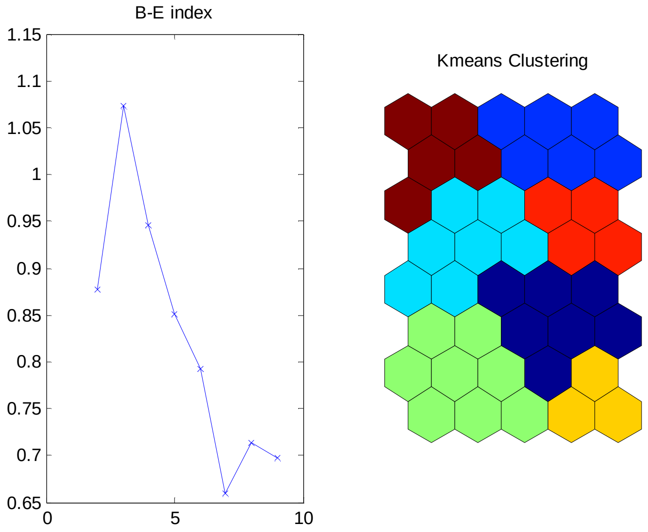


Figure 8-9. *k*-means clustering analysis and visualization on SOM map

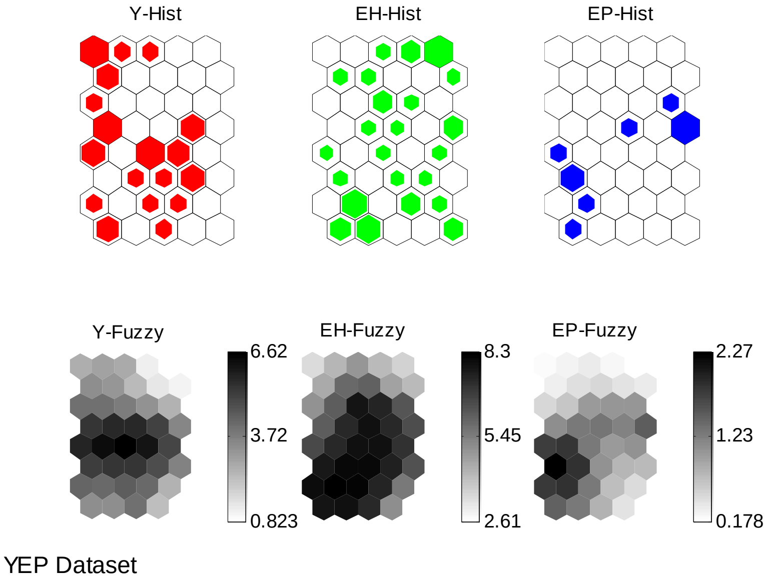


Figure 8-10. Histogram distributions of three gait categories of MFC dataset

The discovered pattern-specific gait locality could be used as a diagnostic measure for gait analysis in clinical applications. For example, we label the map units of gait patterns illustrated in Figure 8-9 with the grades in a range from 1, representing the most healthy gait pattern, to 6 standing for the most impaired gait pattern, by incorporating the gait locality analysis. Then, we utilize the graded labels to diagnose the new subject’s gait status, that is, we determine the possible locality of new subject’s gait on the SOM by detecting the belongingness of new subject’s gait to the discovered gait patterns. If the new subject’s gait is fitting into the cluster with the smaller grade, the subject might walk in a normal manner; conversely, the affiliation the cluster with the bigger grade might indicate the worse walking performance of the new subject. This graphic illustration will provide a useful assistant to gait clinician or researchers in practical applications.

8.4. Related work

Gait pattern analysis has been addressed to reveal kinematic, kinetic and electromyographic (EMG) gait characteristic for modelling human walking [109-111]. The discovery of gait pattern will help to identify any changes of gait that reflects the gait degeneration due to various pathological reasons. One of its applications is to monitor the ageing influence on gait pattern, which causes constant threats to the elderly population and help to prevent them from potential risks of fall. There are various types of gait variables that are used to describe and analyse gait. However, basic gait variables (e.g. walking speed, stride length, cadence, leg length etc), are frequently employed in modelling human walking [112]. As a result of gait pattern analysis, there is a increased demand to identify the related subsets of these gait parameters as feature vector (i.e. gait data model) and employ applicable statistical analysis tools on the derived data model to reveal the underlying gait patterns hidden in the gait data.

To characterize gait patterns and differentiate the normal from the pathological, some pattern recognition and machine learning techniques have been used to address this problem. Some academics utilize supervised approaches [113, 114], in which groups of subjects are defined by *a priori*, to model the quantitative correlation among them by using discriminant analysis paradigm, while others exploit descriptive or subjective techniques to build up collections of subjects [115, 116]. Alternatively, unsupervised (or weak supervised) techniques are also considered as effective paradigms for gait pattern recognition. [102] proposed a fuzzy clustering technique employed on temporal-distance parameters to group normal and pathological gait subjects into various clusters accordingly. *Neural Network* (*NN*) and *Support Vector Machine* (*SVM*), two types of well-studied machine learning approaches recently are used for identification of the at-risk gait in the elderly population. The former adopts neural network model to classify various gait types, while the latter is to recognize gait patterns by finding an optimal separating hyperplane to separate two groups' data. For example, [117, 118] applied *NN* on the selected feature subset from lower-limb joint-angle measures to differentiate various gait pattern, whereas [119] exploited *Minimum Foot Clearance* (MFC) histogram-plot and Poincaré-plot images to train *SVM*, for automated recognition of gait pattern changes due to ageing. Results from their work have shown they are effective gait analysis tools for solving classification problems by learning gait data with satisfactory performance [53-55]. Furthermore, the differentiated subject groups will provide biomechanical insights and treatment assessment criterion for the population with pathological gait characteristics. However, most of the above studies mainly focused on the classification of the subject gait data into predefined subject groups rather than the discovery of underlying relationships among gait data that is used to derive gait patterns as well as identification of subject groups (i.e. gait patterns) from the gait parameters. In most cases, it is crucial to address the issues of how to find a reasonable grouping scheme and then partition the subjects into the corresponding groups since it is hard to define *a priori* subject groups in real scenarios.

8.5. Conclusion

Scientific gait (walking) analysis provides valuable information about an individual's locomotion function, in turn, assists to undertake appropriate measures for clinical diagnosis and prevention, such as assessing treatment for patients with impaired postural control and detecting risk of fall in elderly population.

In this chapter, we have extended the methodologies developed in Web usage mining, to address gait pattern mining for clinical movement diagnosis via user profiling techniques. Upon the constructed gait variable space, we aim to apply clustering-based learning techniques to discover subject gait clusters, which are representing various human walking characteristics, such as normal or pathological gait patterns. In particular, standard and SOM-based clustering algorithms are proposed to perform gait pattern mining on two gait datasets of Cerebral Palsy and elderly population, respectively. The experimental results have demonstrated that the proposed approaches are capable of well partitioning gait data into a number of gait clusters that are corresponding to various gait statuses. And the discovered gait characteristics in the forms of gait profiles or visualization map will provide a promising means for gait analysis in gait clinical application and research.

9. Conclusions

9.1. Summary

With the rapid development of Web applications and great flux of Web information available on the Internet, World Wide Web has become very popular recently and brought us a powerful platform to disseminate information and retrieve information as well as analyse information. Although the progress of the web-based data management research results in developments of many useful Web applications or services, like Web search engines, users are still facing the problems of information overload and drowning due to the significant and rapid growth in the amount of information and the number of users. In particular, Web users usually suffer from the difficulties of finding desirable and accurate information on the Web due to two problems of low precision and low recall caused by the above reasons. Thus, the emerging of Web has put forward a lot of challenges to Web researchers for web-based information management and retrieval.

Web mining could be partly used to solve the problems mentioned above directly or indirectly. In principle, Web mining techniques are the means of utilizing data mining methods to induce and extract useful information from Web information and service. Practically, Web mining could be classified into three categories: *Web content mining, Web structure mining*, and *Web usage mining*. This dissertation concentrates on Web usage mining.

Web recommendation or personalization could be viewed as a process that recommends the customized Web presentations or predicts tailored Web contents to users according to their specific taste or preference. There are two kinds of approaches and techniques commonly used in Web recommendation, namely content-based filtering and collaborative filtering systems. Nowadays, Web usage mining has been proposed as an alternative method for not only revealing user access patterns, but also making Web recommendations.

On the other hand, *Latent Semantic Analysis* (LSA) is an approach to capture the latent or hidden semantic relationships among co-occurrence activities, which has been widely used in information indexing and retrieval applications. Despite of the considerable progress of the traditional LSA approach, it still has some shortcomings, such as computational difficulty of sparsity problem of co-occurrence matrix, overfitting problem, capability of capturing latent semantic space etc. To address these, some studies have extended the standard LSA techniques via introducing various statistical background principles, such as PLSA and LDA models.

In this study, we have addressed Web usage mining for Web recommendation by using latent semantic analysis paradigms. This dissertation mainly focuses on discovering Web usage patterns in terms of task-oriented Web user profiles and Web page groups from Web log files to support Web recommendations via various latent semantic analysis (LSA) paradigms, the main strengths of the employed LSA-based techniques are the capabilities of finding the underlying relationships among the web objects and identifying the latent navigational tasks associated. The experimental results conducted on three real world datasets have verified the effectiveness of finding better quality Web object clusters and shown the improvement of recommendation accuracy compared to other existing recommendation algorithms. Among the three LSA-based algorithms, LDA model demonstrates the best recommendation rate with less computational costs. Another interesting investigation is implemented by extending the developed methodologies and algorithms to another data mining application, i.e. a healthcare data mining application. By modelling the gait feature vector, we conduct clustering analysis for gait pattern mining. The discovered gait profiles in the form of weighted gait variable vectors provide an assisted diagnostic means for monitoring the changes of gait statuses due to various physical or pathological reasons.

In detail, to achieve these goals a mathematical framework is established for Web usage mining and a series of algorithms are proposed to predict Web user navigational preferences and recommend the customized Web contents to Web user. Three kinds of latent semantic analysis models, namely standard LSA, PLSA and LDA, are proposed to address Web usage mining and Web recommendation respectively. Two case studies of extension of the proposed pattern mining methodologies and algorithms are carried out in the application of gait pattern mining, which is one important topic in healthcare and biomechanical data mining, to assess the effectiveness and efficiency of the proposed techniques.

9.1.1. Mathematical Framework of Web Usage Mining for Web Recommendation

This framework is based on the matrix theory in linear algebra. In this framework, Web usage data is characterized in a matrix model, in which each element reflects the navigational preference of Web users. This framework makes it feasible to systematically perform analysis on the collected Web usage data using mathematical theories in a unified way that leads to extending the developed methodologies to other data mining applications, which have similar data expressions. As a result, this framework provides a solid mathematical base for discovering Web usage patterns and making Web recommendations.

9.1.2. Latent Semantic Analysis Models for Web Usage Mining

In this dissertation we have intensively investigated using latent semantic analysis paradigms for discovering Web usage patterns and making Web recommendations, which includes the following analytical models:

- Traditional Latent Semantic Indexing (LSI)
- Probabilistic Latent Semantic Analysis (PLSA)
- Latent Dirichlet Allocation (LDA)

Tradition latent semantic indexing is based on Singular Value Decomposition (SVD) operation, which is to reduce the dimensionality of the original input space but holding the maximum approximation of the original matrix. The main advantage of LSI model is its capability of uncovering underlying relationships among the observed objects that aren't exhibited explicitly and directly. In this study, we aim to employ the intuitive LSI analysis on the usage data matrix to analyse the association between user sessions in a transformed vector space resulted from a SVD implementation.

Probabilistic Latent Semantic Analysis (PLSA) model is a variant of the tradition LSI model, which introduces an aspect space as an inter-medium between two usage attributes, i.e. user session and Web page. With PLSA model, the original usage data is mapped into two new usage vectors, in which the associations between user sessions and latent factors, and between Web pages and latent factors, are modelled by the estimates of conditional probabilities. The new mapped usage vectors along with the newly defined user session similarity and Web page similarity provide a novel Web usage mining way, with which we can derive usage based page clusters and session aggregates (or user profiles).

Latent Dirichlet Allocation (LDA) is a recently emerging generative model, which reveals the intrinsic correlation among co-occurrence via a generative procedure. In contrast to mining Web usage pattern by PLSA model, LDA is to learn hidden usage knowledge based on computing the Dirichlet value and posterior probability. The discovered usage knowledge is then used to predict user's potentially interested Web contents. The common strength of the latter two models is the capability of capturing the aspect space that associates with the discovered usage knowledge in addition to usage pattern mining itself

9.1.3. Algorithms for Web Usage Mining and Web Recommendation

Upon the proposed data analysis models, we have developed a number of algorithms for Web usage mining and Web recommendation. To achieve this, we also introduced a series of concepts or definitions to model the relationships among Web objects.

The developed algorithms for Web usage mining and Web recommendation could be summarized as follows:

- SVD and EM algorithms for capturing the underlying association hidden in usage data. These learned usage knowledge using the above algorithms is expressed in a unified feature vector.

- Clustering and probability inference algorithms for user profiling. With these algorithms, user sessions are aggregated into a number of session clusters, which are used to generate usage patterns in terms of the centroids of the clusters, and a number of page categories, which are considered as page functional aggregates.
- Latent semantic factor space analysis algorithms. These algorithms are to extract the latent task space by selecting and interpreting the significant Web pages that greatly contributed to the corresponding latent tasks.
- Usage based recommendation algorithms for Web recommendations. These kinds of algorithms aim to make use of the usage knowledge derived from Web usage mining for Web recommendation. It is mainly on a basis of the collaborative filtering algorithm, which is to make recommendations via referring to other's visiting preferences that have similar access preference. A top-N weighted scoring scheme is proposed to form the core part of scoring in the recommendation framework.

To evaluate the cluster quality and recommendation accuracy, we have also adopted two metrics and a baseline measure to compare the performance and effectiveness of the proposed data analysis models and algorithms.

9.1.4. Case Studies of Gait Pattern Mining

In this dissertation, we have also investigated the extension of the developed methodologies to gait pattern mining. Gait analysis is an important topic in movement clinical research and an application for specific populations, such as CP patients or elderly people. In this study, we conduct the following case studies:

- Traditional clustering case study for CP gait pattern mining. We develop the k-means and hierarchical clustering based approaches to find CP-specific gait patterns. From the experiments, it is shown that the gait characteristics of healthy children and CP patients at different pathological levels are substantially separated into different groups and the discovered gait pattern knowledge can provide a helpful means for researchers or clinician to monitor the development of CP or assess the effectiveness of the intervention.
- SOM-based clustering case study for monitoring fall risks of elderly population. We employ a SOM-based clustering algorithm to investigate the locality of the gait pattern in a transformed SOM grid. The experiments on the gait data of three subject groups have demonstrated that the derived SOM grid may potentially give us a visualized representation for screening the gait status and monitoring fall risk in elderly population.

9.2. Possible Future work

In this dissertation, we have concentrated on the research of Web usage mining for Web recommendation via latent semantic analysis paradigms. The theoretical and experimental studies have shown the effectiveness and applicability of the proposed models and approaches.

The future work can be continued along the following directions:

- Integration of ontology knowledge of Web pages into Web recommendation. The current research is mainly based on analysis of Web usage knowledge, not taking other Web data sources into account. With the development of semantic Web and ontology research, it is believed that ontology knowledge of Web pages can provide deeper understanding or semantic linking of Web pages as a result of conveying the conceptual information. Ontology knowledge could be viewed as a high-level knowledge representation over the intuitive content knowledge. Hence, integrating the ontology knowledge with the usage knowledge will substantially improve the accuracy and efficiency of Web recommendation.

- Employing the latest progress of other related research areas into Web data management. The successes and contributions from data mining, machine learning, information retrieval domains always bring in new data models and algorithms to Web data research. It is believed these progresses will produce a big potential for Web researchers to address the open research problems not solved yet.
- Expanding the scope of current research to other related areas. Web data mining and community analysis on Web pages or users provides an interesting and promising way to discover the aggregation nature of co-occurrence based on statistical learning approaches. With the emerging of new applications over the Internet, especially Web 2.0 technology, many new types of Web data, such as email traffic, web-blog, and wiki pages are available. These data types have produced a large amount of new knowledge resources, which leads to new research directions, for example, social network analysis.

Reference

1. Kosala, R. and H. Blockeel, *Web Mining Research: A Survey*. SIGKDD Explorations, 2000. 2(1): p. 1-15.
2. Zhang, Y., J.X. Yu, and J. Hou, *Web Communities: Analysis and Construction*. 2006, Berlin Heidelberg: Springer.
3. Ghani, R. and A. Fano. *Building Recommender Systems Using a Knowledge Base of Product Semantics*. in *Proceedings of the Workshop on Recommendation and Personalization in E-Commerce, at the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems (AH2002)*. 2002, p. 11-19, Malaga, Spain.
4. Chakrabarti, S., et al. *The Structure of Broad Topics on the Web*. in *Proceeding of 11th International World Wide Web Conference*. 2002, p. 251 - 262, Honolulu, Hawaii, USA.
5. Büchner, A.G. and M.D. Mulvenna, *Discovering Internet Marketing Intelligence through Online Analytical Web Usage Mining*. SIGMOD Record, 1998. 27(4): p. 54-61.
6. Chang, G., et al., eds. *Mining the World Wide Web: An Information Search Approach*. The Information Retrieval. Vol. 10. 2001, KAP.
7. Pierrakos, D., et al. *Web Community Directories: A New Approach to Web Personalization*. in *Proceeding of the 1st European Web Mining Forum (EWMF'03)*. 2003, p. 113-129, Cavtat-Dubrovnik, Croatia.
8. Mobasher, B., *Web Usage Mining and Personalization*, in *Practical Handbook of Internet Computing*, M.P. Singh, Editor. 2004, CRC Press. p. 15.1-37.
9. *http://searchenginewatch.com/*.
10. Brin, S. and L. Page, *The PageRank Citation Ranking: Bringing Order to the Web (http://www-db.stanford.edu/~backrub/pageranksub.ps.)*. 1998.
11. Ding, C., et al., *PageRank, HITS and a Unified Framework for Link Analysis*, L.B.N.L.T. Report, Editor. 2002, University of California, Berkeley, CA.
12. 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.
13. Haveliwala, T. *Topic-Sensitive PageRank*. in *Proceedings of the 11th International World Wide Web Conference*. 2002, p. 517-526, Honolulu, Hawaii, USA.
14. Kamvar, S., H. TH, and G.G. Manning CD. *Extrapolation Methods for Accelerating PageRank Computations*. in *Proceedings of WWW'03*. 2003, p. 261-270, Budapest, Hungary.

15. Page, L., Brin S, Motwani R, Winograd T, *The Pagerank Citation Ranking: Bringing Order to the Web*, in *Report*. 1998, Report in Computer Science Department, Stanford University.
16. Richardson, M. and D. P. *The Intelligent Surfer:Probabilistic Combination of Link and Content Information in PageRank*. in *2001 Neural Information Processing Systems Conference (NIPS 2001)*. 2001, p. 1441-1448, Vancouver, British Columbia, Canada: MIT Press, Cambridge, MA.
17. Han, J. and M. Kamber, *Data Mining: Concepts and Techniques*. 2007: Morgan Kaufmann.
18. Borges, J., Levene M, *Data Mining of User Navigation Patterns.* Web Usage Analysis and User Profiling, 2000. 1836: p. 92-111.
19. Agrawal, R. and R. Srikant. *Mining Sequential Patterns*. in *Proceedings of the International Conference on Data Engineering (ICDE)*. 1995, p. 3-14, Taipei, Taiwan: IEEE Computer Society Press.
20. Kumar, R., et al. *Extracting large-scale knowledge bases from the web*. in *Proceeding of the 25th VLDB Conference*. 1999, p. 639-650, Edinburgh, Scotland.
21. Lang, K. *Weeder: Learning to Filter Netnews*. in *Proceeding of 12th International Conference on Machine Learning ICML95*. 1995, p. 331-339, Tahoe City, California, USA.
22. Liu, B. and K.C.-C. Chang, *Special Issue on Web Content Mining.* ACM SIGKDD Explorations, 2004. 6(2): p. 1-4.
23. Srivastava, J., et al., *Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data.* ACM SIKDD Explorations, 2000. 1(2): p. 12-23.
24. Chakrabarti, S., *Data mining for hypertext: A Tutorial Survey.* ACM SIGKDD Explorations Newsletter, 2000. 1(2): p. 1-11.
25. Hou, J. and Y. Zhang, *Effectively Finding Relevant Web Pages from Linkage Information.* IEEE Trans. Knowl. Data Eng., 2003. 15(4): p. 940-951.
26. Kleinberg, J. *Authoritative Sources in a Hyperlinked Environment*. in *Proceeding of 9th ACM-SIAM Symposium on Discrete Algorithms*. 1998, p. 668-677, San Francisco, California.
27. Craven, M., et al. *Learning to Extract Symbolic Knowledge From the World Wide Web*. in *Proceedings of the Fifteenth National Conference on Artificial Intellligence (AAAI'98)*. 1998, p. 509-516, Madison, Wisconsin, USA.
28. Srivastava, J., et al., *Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data.* SIGKDD Explorations, 2000. 1(2): p. 12-23.
29. Mobasher, B., et al., *Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization.* Data Mining and Knowledge Discovery, 2002. 6(1): p. 61-82.
30. Madria, S.K., et al. *Research Issues in Web Data Mining*. in *Proceedings of Data Warehousing and Knowledge Discovery, First International Conference, DaWaK '99*. 1999, p. 303-312, Florence, Italy.

31. Tan, A.-H. *Text mining: The state of the art and the challenges*. in *Proceednngs of the Pacific Asia Conf on Knowledge Discovery and Data Mining PAKDD'99 workshop on Knowledge Discovery from Advanced databases*. 1999, p. 65-70, Florence, Italy.
32. Gibson, D., J. Kleinberg, and P. Raghavan. *Inferring Web Communities from Link Topology*. in *Proceeding of 9th ACM Conference on Hypertext and Hypermedia*. 1998, p. 225-234, Pittsburgh, USA.
33. Kumar, R., et al. *Trawling the Web For Emerging Cyber-Communities*. in *Proceedings of the 8th International World Wide Web Conference*. 1999, p. 1481-1493, Toronto, Canada.
34. Hou, J. and Y. Zhang. *Constructing Good Quality Web Page Communities*. in *Proc. of the 13th Australasian Database Conferences (ADC2002)*. 2002, p. 65-74, Melbourne, Australia: ACS Inc.
35. 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.
36. Shahabi, C., et al. *Knowledge Discovery from User Web-Pge Nvigational*. in *Proceedings of the 7th International Workshop on Research Issues in Data Engineering (RIDE '97)* 1997, p. 20-29: IEEE Computer Society.
37. Zhou, Y., X. Jin, and B. Mobasher. *A Recommendation Model Based on Latent Principal Factors in Web Navigation Data*. in *Proceedings of the 3rd International Workshop on Web Dynamics*. 2004, New York: ACM Press.
38. Dunja, M., *Personal Web Watcher: Design and Implementation (Report)*. 1996, Department of Intelligent Systems, J. Stefan Institute, Slovenia.
39. Herlocker, J.L., et al., *Evaluating Collaborative Filtering Recommender Systems.* ACM Transaction on Information Systems (TOIS), 2004. 22(1): p. 5 - 53
40. Joachims, T., D. Freitag, and T. Mitchell. *Webwatcher: A Tour Guide For the World Wide Web*. in *The 15th International Joint Conference on Artificial Intelligence (IJCAI'97)*. 1997, p. 770-777, Nagoya, Japan.
41. Lieberman, H. *Letizia: An Agent that Assists Web Browsing*. in *Proc. of the 1995 International Joint Conference on Artificial Intelligence*. 1995, p. 924-929, Montreal, Canada: Morgan Kaufmann.
42. Herlocker, J., et al. *An Algorithmic Framework for Performing Collaborative Filtering*. in *Proceedings of the 22nd ACM Conference on Researchand Development in Information Retrieval (SIGIR'99)*. 1999, p. 230-237, Berkeley, CA, USA.
43. Konstan, J., et al., *Grouplens: Applying Collaborative Filtering to Usenet News.* Communications of the ACM, 1997. 40: p. 77-87.
44. Shardanand, U. and P. Maes. *Social Information Filtering: Algorithms for Automating 'Word of Mouth'*. in *Proceedings of the Computer-Human Interaction Conference (CHI95)*. 1995, p. 210-217, Denver, Colorado.

45. Han, E., et al., *Hypergraph Based Clustering in High-Dimensional Data Sets: A Summary of Results*. IEEE Data Engineering Bulletin, 1998. 21(1): p. 15-22.
46. 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.
47. Agarwal, R., C. Aggarwal, and V. Prasad, *A Tree Projection Algorithm for Generation of Frequent Itemsets*. Journal of Parallel and Distributed Computing 1999. 61(3): p. 350-371.
48. Agrawal, R. and R. Srikant. *Jorge B. Bocca and Matthias Jarke and Carlo Zaniolo*. in *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*. 1994, p. 487-499, Santiago, Chile: Morgan Kaufmann.
49. Mobasher, B., R. Cooley, and J. Srivastava. *Creating Adaptive Web Sites Through Usage-Based Clustering of URLs*. in *Proceedings of the 1999 Workshop on Knowledge and Data Engineering Exchange* 1999, p. 19-25: IEEE Computer Society.
50. Ngu, D.S.W. and X. Wu. *Sitehelper: A Localized Agent that Helps Incremental Exploration of the World Wide Web*. in *Proceedings of 6th International World Wide Web Conference*. 1997, p. 1249-1255, Santa Clara, CA,: ACM Press.
51. Cohen, E., B. Krishnamurthy, and J. Rexford. *Improving End-To-End Performance of the Web Using Server Volumes and Proxy ltems*. in *Proceedings of the ACM SIGCOMM'98*. 1998, p. 241-253, Vancouver, British Columbia, Canada ACM Press.
52. Perkowitz, M. and O. Etzioni. *Adaptive Web Sites: Conceptual Cluster Mining*. in *Proceeding of 16th International Joint Conference on Articial Intelligence*. 1999, p. 264-269, Stockholm, Sweden: Morgan Kaufmann.
53. Lee, L. and W.E.L. Grimson. *Gait Analysis for Recognition and Classification*. in *Proceedings 5th International Conference Automatic Face Gesture Recognition (FGR'02)*. 2002, p. 148-155, Washington, DC, USA.
54. Chapelle, O., P. Haffner, and V.N. Vapnik, *Support Vector Machines for Histogram-Based Classification*. IEEE Trans. Neural Netw., 1999. 10(5): p. 1055-1064.
55. Chan, K., et al., *Comparison of Machine Learning and Traditional Classifiers in Glaucoma Diagnosis*. IEEE Trans. Biomed. Eng., 2002. 49(9): p. 963-974.
56. Xiao, J., et al. *Measuring Similarity of Interests for Clustering Web-Users*. in *Proceedings of the 12th Australasian Database conference (ADC2001)*. 2001, p. 107-114, Queensland, Australia: ACS Inc.
57. Berendt, B., A. Hotho, and G. Stumme. *Towards Semantic Web Mining*. in *Proceedings of the First International Semantic Web Conference (ISWC02)*. 2002, p. 264-278, Sardinia, Italy.
58. Mitchell, T., et al., *Experience with a Learning Personal Assitant*. Communications of the ACM, 1994. 37(7): p. 81-91.

59. Pazzani, M., J. Muramatsu, and D. Billsus. *Syskill & Webert: Identifying interesting Web sites*. in *AAAI Spring Sysmposium on Machine Learning in Information Access and Proceedings of 3th National Conference on Artificial Intelligence AAAI 96*. 1996, p. 54-61, Stanford, USA.
60. Oberle, D., *Semantic Community Web Portals Personalization (Report)*. 2000, Universit¨at Karlsruhe.
61. Baeza-Yates, R. and B. Ribeiro-Neto, *Modern Information Retrieval*. 1999: Addison Wesley, ACM Press.
62. Deerwester, S., et al., *Indexing by latent semantic analysis.* Journal American Society for information retrieval, 1990. 41(6): p. 391-407.
63. 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.
64. Berry, M.W., S.T. Dumais, and G.W. O' Brie0146-4833n, *Using linear algebra for intelligent information retrieval.* SIAM Review, 1995. 37(4): p. 573-595.
65. 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 Mining and Applications (ADMA 2005)*. 2005, p. 31-42, Wuhan, china: Springer.
66. Hofmann, T. *Probabilistic Latent Semantic Analysis*. in *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*. 1999, p. 50-57, Berkeley, California, USA: ACM Press.
67. Blei, D.M., A.Y. Ng, and M.I. Jordan, *Latent Dirichlet Allocation.* Journal of Machine Learning Research, 2003(3): p. 993-1022.
68. Asano, Y., et al. *Finding Neighbor Communities in the Web Using Inter-site Graph*. in *Proc. of the 14th International Conference on Database and Expert Systems Applications (DEXA'03)*. 2003, p. 558-568, Prague, Czech Republic.
69. Flake, G.W., R.E. Tarjan, and K. Tsioutsiouliklis, *Graph Clustering and Minimum Cut Trees.* Internet Mathematics, 2004. 1(4).
70. Tawde, V.B., T. Oates, and E. Glover. *Generating Web Graphs with Embedded Communities*. in *Proceedings of 3rd Workshop on Algorithms and Models for the Web-Graph (WAW 2004)*. 2004, p. 80-91, Rome, Italy.
71. Pierrakos, D., et al. *Construction of Web Community Directories by Mining Usage Data*. in *Proceeding of the 2nd Hellenic Data Management Symposium (HDMS'03)*. 2003, Athens, Greece.
72. Otsuka, S., et al. *Extracting User Behavior by Web Communities Technology on Global Web Logs*. in *Proc. of the 15th International Conference on Database and Expert Systems Applications (DEXA'04)*. 2004, p. 957-988, Zaragoza, Spain.
73. Cooley, R., B. Mobasher, and J. Srivastava, *Data Preparation for Mining World Wide Web Browsing Patterns.* Journal of Knowledge and Information Systems, 1999. 1(1): p. 5-32.

74. Broder, A., et al. *Syntactic Clustering of the Web*. in *Proceedings of the 6th International WWW Conference*. 1997, p. 391-404, Santa Clara, CA, USA.
75. 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.
76. O'Conner, M. and J. Herlocker. *Clustering Items for Collaborative Filtering*. in *Proceedings of the ACM SIGIR Workshop on Recommender Systems*. 1999, Berkeley, CA, USA: ACM Press.
77. Datta, B.N., *Numerical Linear Algebra and Application*. 1995: Brooks/Cole Publishing Company.
78. Hofmann, T., *Latent Semantic Models for Collaborative Filtering.* ACM Transactions on Information Systems, 2004. 22(1): p. 89-115.
79. Sarwar, B.M., et al. *Item-based collaborative filtering recommendation algorithms*. in *Proceedings of the 10th International World Wide Web Conference (WWW10).* 2001, p. 285-295, Hong Kong.
80. Jin, X. and B. Mobasher. *Using Semantic Similarity to Enhance Item-Based Collaborative Filtering*. in *Proceedings of The 2nd International Conference on Information and Knowledge Sharing*. 2003, p. 85-96, Scottsdale, Arizona.
81. Perkowitz, M. and O. Etzioni, *Adaptive Web sites.* Communications of the ACM, 2000. 43(8): p. 152 - 158.
82. Xu, G., et al. *Discovering User Access Pattern Based on Probabilistic Latent Factor Model*. in *Proceeding of 16th Australasian Database Conference*. 2005, p. 27-36, Newcastle, Australia: ACS Inc.
83. Xu, G., Y. Zhang, and X. Zhou. *Using Probabilistic Semantic Latent Analysis for Web Page Grouping*. in *15th International Workshop on Research Issues on Data Engineering: Stream Data Mining and Applications (RIDE-SDMA'2005)*. 2005, p. 29-36, Tokyo, Japan.
84. Jin, X., Y. Zhou, and B. Mobasher. *A Unified Approach to Personalization Based on Probabilistic Latent Semantic Models of Web Usage and Content*. in *Proceedings of the AAAI 2004 Workshop on Semantic Web Personalization (SWP'04)*. 2004, San Jose.
85. Dempster, A.P., N.M. Laird, and D.B. Rubin, *Maximum Likelihood from Incomplete Data via the EM Algorithm.* Journal Royal Statist. Soc. B, 1977. 39(2): p. 1-38.
86. Hofmann, T., *Unsupervised Learning by Probabilistic Latent Semantic Analysis.* Machine Learning Journal, 2001. 42(1): p. 177-196.
87. Cohn, D. and H. Chang. *Learning to Probabilistically Identify Authoritative Documents*. in *Proceedings of the 17th International Conference on Machine Learning*. 2000, p. 167-174, San Francisco, CA: Morgan Kaufmann

88. Cohn, D. and T. Hofmann, *The Missing Link: A Probabilistic Model of Document Content and Hypertext Connectivity*, in *Advances in Neural Information Processing Systems 13*, T.K. Leen, T.G. D., and V. Tresp, Editors. 2001, MIT Press.
89. Elango, P.K. and K. Jayaraman, *Clustering Images Using the Latent Dirichlet Allocation Model (http://pages.cs.wisc.edu/~pradheep/Clust-LDA.pdf).* 2005.
90. Song, Y., et al. *Efficient Topic-based Unsupervised Name Disambiguation*. in *Joint Conference in Digital Library 2007*. 2007, p. 342-351, Vancouver, British Columbia, Canada.
91. Ma, J., Y. Zhang, and J. Cao. *A Probabilistic Semantic Approach for Discovering Web Services*. in *Proceedings of WWW2007*. 2007, p. 1221-1222, Banff, Alberta, Canada.
92. Wei, X. and W.B. Croft. *LDA-Based Document Models for Ad-hoc Retrieval*. in *Proceedings of SIGIR'06* 2006, p. 178-185, Seattle, Washington, USA.
93. Li, F.-F. and P. Perona. *A Bayesian Hierarchical Model for Learning Natural Scene Categories*. in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 2005, p. 524 - 531, San Diego, CA, USA.
94. Wang, X. and A. McCallum. *Topics over Time: A NonMarkov ContinuousTime Model of Topical Trends*. in *Proceedings of ACM SIGKDD*. 2006, p. 424-433, Philadelphia, Pennsylvania, USA.
95. Russell, S.J. and P. Norvig, *Artificial Intelligence, A Modern Approach*. 1995: Prentice Hall.
96. Xu, G., Y. Zhang, and X. Zhou. *A Web Recommendation Technique Based on Probabilistic Latent Semantic Analysis*. in *Proceeding of 6th International Conference of Web Information System Engineering (WISE'2005)*. 2005, p. 15-28, New York City, USA: LNCS 3806.
97. Liu, K., et al., *Client-side Web Mining for Community Formation in Peer-to-Peer Environments.* ACM SIGKDD Explorations Newsletter, 2006. 8(2): p. 11 - 20.
98. Bose, A., et al., *Incorporating Concept Hierarchies into Usage Mining Based Recommendations.* Advances in Web Mining and Web Usage Analysis, 2007: p. 110--126.
99. Rashid, A.M., et al. *ClustKNN: A Highly Scalable Hybrid Model- & Memory-Based CF Algorithm*. in *Proceedings of WEBKDD'06*. 2006, Philadelphia, Pennsylvania, USA.
100. Jin, X., Y. Zhou, and B. Mobasher. *A Maximum Entropy Web Recommendation System: Combining Collaborative and Content Features*. in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'05)*. 2005, p. 612-617, Chicago.
101. Inman, V.T., H.J. Ralston, and F. Todd, *Human Walking*. 1981, Baltimore, MD: Williams and Wilkins.

102. O’Malley, M.J., et al., *Fuzzy Clustering of Children with Cerebral Palsy Based on Temporal-Distance Gait Parameters.* IEEE Trans. ON Rehab. Eng. , 1997. 5(4): p. 300-309.
103. Xu, G., Y. Zhang, and X. Zhou. *Towards User Profiling for Web Recommendation*. in *The 18th Australian Joint Conference on Artificial Intelligence (AI'2005)*. 2005, p. 405-414, Sydney, Australia: LNAI 3809, Springer.
104. Hotho, A., A. Mädche, and S. Staab. *Ontology-based Text Clustering*. in *Workshop "Text Learning: Beyond Supervision", IJCAI 2001*. 2001.
105. Begg, R., M. Palaniswami, and B. Owen, *Support Vector Machines for Automated Gait Recognition.* IEEE Transactions on Biomedical Engineering, 2005. 52: p. 828-838.
106. Vesanto, J. and E. Alhoniemi, *Clustering of the Self-Organizing Map.* IEEE TRANSACTIONS ON NEURAL NETWORKS 2000. 11(3): p. 586-599.
107. Barton, G.J., *Visualisation of Gait Data Using A Self Organising Artificial Neural Network*, in *Computational Intelligence for Movement Sciences: Neural Networks, Support Vector Machines and other Emerging Techniques*, R.a.P. Begg, M, Editor. 2006, Idea Group Inc., USA. p. 197-216.
108. Smith, K.A. and A. Ng, *Web Page Clustering using A Self-Organizing Map of User Navigation Patterns.* Decision Support Systems, 2003. 35: p. 245- 256.
109. Vaughan, C.L., B.L. Davis, and J.C. O’Connor, *Dynamics of Human Gait*, in *Human Kinetics*. 1992: Champaign, IL.
110. Judge, J.O., R.B. Davis, and S. O˘ unpuu, *Step Length Reductions in Advanced Age: The Role of Ankle and Hip Kinetics.* J. Gerontol.: Med. Sci., 1996. 51: p. 303-312.
111. Nigg, B.M., V. Fisher, and J.L. Ronsky, *Gait Characteristics as a Function of Age and Gender.* Gait Posture, 1994. 2: p. 213-220.
112. Ostrosky, K.M., et al., *A Comparison of Gait Characteristics in Young and Old Subjects.* Phys. Ther., 1994. 74: p. 637-646.
113. Tibarewala, D.N. and S. Ganguli, *Pattern Recognition in Tachographic Gait Records of Normal and Lower Extremity Handicapped Human Subjects.* J. Biomed. Eng., 1982. 4: p. 233-240.
114. Damiano, D.L. and M.F. Abel, *Relationship of Gait Analysis to Gross Motor Function in Cerebral Palsy*. Develop. Med. Child Neurol., 1996. 38: p. 389-396.
115. Winters, T.F., J.G. Gage, and R. Hicks, *Gait Patterns in Spastic Hemiplegia in Children and Young Adults.* J. Joint Bone Surg, 1987. 69A: p. 437-441.
116. Perry, J., et al., *Classification of Walking Handicap in the Stroke Population.* Stroke, 1995. 26: p. 982-989.
117. Barton, J.G. and A. Lees, *An Application of Neural Networks for Distinguishing Gait Patterns on the Basis of Hip-Knee Joint Angle Diagrams.* Gait Posture, 1997. 5: p. 28-33.
118. Holzreiter, S.H. and M.E. Kohle, *Assessment of Gait Pattern Using Neural Networks.* J. Biomech., 1993. 26: p. 645-651.

119. Begg, R.K., M. Palaniswami, and B. Owen, *Support Vector Machines for Automated Gait Classification.* IEEE Tran. on Biomed. Eng., 2005. 52(5).