University of Huddersfield

University of Huddersfield Repository

Saad, Elmak

Optimizing E-Management Using Web Data Mining

Original Citation

Saad, Elmak (2018) Optimizing E-Management Using Web Data Mining. Doctoral thesis, University of Huddersfield.

This version is available at http://eprints.hud.ac.uk/id/eprint/34540/

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

http://eprints.hud.ac.uk/

OPTIMIZING E-MANAGEMENT USING WEB DATA MINING

ELMAK ELMASSAD SAAD

A thesis submitted to the University of Huddersfield
in partial fulfilment of the requirements for
the degree of Doctor of Philosophy

School of Computing and Engineering
University of Huddersfield

March 2017

Copyright statement

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns any copyright in it (the “Copyright”) and s/he has given the University of Huddersfield the right to use such Copyright for any administrative, promotional, educational and/or teaching purposes.

ii. Copies of this thesis, either in full or in extracts, may be made only in accordance with the regulations of the University Library. Details of these regulations may be obtained from the Librarian. This page must form part of any such copies made.

iii. The ownership of any patents, designs, trade marks and any and all other intellectual property rights except for the Copyright (the “Intellectual Property Rights”) and any reproductions of copyright works, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property Rights and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property Rights and/or Reproductions.

Abstract

Today, one of the biggest challenges that E-management systems face is the explosive growth of operating data and to use this data to enhance services. Web usage mining has emerged as an important technique to provide useful management information from user's Web data. One of the areas where such information is needed is the Web-based academic digital libraries. A digital library (D-library) is an information resource system to store resources in digital format and provide access to users through the network. Academic libraries offer a huge amount of information resources, these information resources overwhelm students and makes it difficult for them to access to relevant information. Proposed solutions to alleviate this issue emphasize the need to build Web recommender systems that make it possible to offer each student with a list of resources that they would be interested in. Collaborative filtering is the most successful technique used to offer recommendations to users. Collaborative filtering provides recommendations according to the user relevance feedback that tells the system their preferences. Most recent work on D-library recommender systems uses explicit feedback. Explicit feedback requires students to rate resources which make the recommendation process not realistic because few students are willing to provide their interests explicitly. Thus, collaborative filtering suffers from “data sparsity” problem. In response to this problem, the study proposed a Web usage mining framework to alleviate the sparsity problem. The framework incorporates clustering mining technique and usage data in the recommendation process. Students perform different actions on D-library, in this study five different actions are identified, including printing, downloading, bookmarking, reading, and viewing the abstract. These actions provide the system with large quantities of implicit feedback data. The proposed framework also utilizes clustering data mining approach to reduce the sparsity problem. Furthermore, generating recommendations based on clusters produce better results because students belonging to the same cluster usually have similar interests.

The proposed framework is divided into two main components: off-line and online components. The off-line component is comprised of two stages: data pre-processing and the derivation of student clusters. The online component is comprised of two stages: building student's profile and generating recommendations. The second stage consists of three steps, in the first step the target student profile is classified to the closest cluster profile using the cosine similarity measure. In the second phase, the Pearson correlation coefficient method is used to select the most similar students to the target student from the chosen cluster to serve as a source of prediction. Finally, a top-list of resources is presented.

Using the Book-Crossing dataset the effectiveness of the proposed framework was evaluated based on sparsity level, and Mean Absolute Error (MAE) regarding accuracy. The proposed framework reduced the sparsity level between (0.07% and 26.71%) in the sub-matrices, whereas the sparsity level is between 99.79% and 78.81% using the proposed framework, and 99.86% (for the original matrix) before applying the proposed framework. The experimental results indicated that by using the proposed framework the performance is as much as 13.12% better than clustering-only explicit feedback data, and 21.14% better than the standard K Nearest Neighbours method. The overall results show that the proposed framework can alleviate the Sparsity problem resulting in improving the accuracy of the recommendations.

ACKNOWLEDGEMENTS

I acknowledge with extreme gratitude to my supervisor Prof. Joan Lu, for the advice she has provided throughout my time as her student. Her attention to every academic precision provided me with the necessary direction for my study.

I also had the support of my co-supervisor Dr Ibrahim Dweib, I thank for his patient guidance.

I would also like to thank all the members of the research office at School of Computing and Engineering.

Very special thanks to my colleagues, Dr Ahmed bin Saleh and Mr Abu Obeida Al-Aqub for their continuous support.

Heartfelt thanks to my sister Hiba for her continuous support.

Special thanks to my wife, my daughter and my sons, they have been a constant source of encouragement.

I am most grateful to my mother and father for their belief in me and their constant prayers.

CONTENTS

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Research overview

The aim of management is to ensure that services are provided in an efficient manner (Guozheng and Rongqiu 2007). E-management is described as a strategic approach to manage an organization that depend on IT for achieving services to respond to customer demand (Guozheng and Rongqiu 2007). As the rapid development of computer technology and network technology, libraries moved from paper-based to digital-based. The aim of this transformation is to improve the traditional library service quality (Chen, Tsai et al. 2008). Libraries that intend to provide such digital services are named digital libraries (Zhu and Wang 2007) . Digital library is an information system, which contains abundant and diverse digital resources provided to users using a variety of technologies (Wang, Xu et al. 2005). The utilization of information technology in the digital library has been a hotspot in the study of computer knowledge organizations and digitization (Zhao, Niu et al. 2014). Information technology effectively provides easier management of data such as seeking and filtering and retrieving information in digital libraries (Spink, Wilson et al. 2002; Porcel and Herrera-Viedma 2009; Krishnamurthy and Balasubramani 2014). Digital libraries have been applied in a lot of contexts, but this study focuses on academic digital libraries in universities. Academic digital libraries provide information resources and services to faculty and students in academic organizations that support learning, teaching and research (Bhide, Heung et al. 2007; Witten, Bainbridge et al. 2010; Jange 2015). Although academic D-library provides a significant amount of digital resources, it also opens a number of challenges in the field of academic D-library management of dealing with the overload information and differentiating useful information from misinformation for users (Morales-del-Castillo, Pedraza-Jiménez et al. 2009; Ambayo 2010; Tejeda-Lorente, Bernabé-Moreno et al. 2014). These challenges present a major area of concern to academic D-libraries users which include the problem of wasting time which may lead to lower satisfaction among users. Thus users using the D-library are reduced (Krishnamurthy and Balasubramani 2014). Traditional academic digital library search information service allows users to query certain keywords to access the resources and return search results purely based on keywords input (Jung 2007). In this scenario, users could hardly acquire resources outside of their own search keywords (Li and Chen 2008). In such systems, users need to have experience in the process of searching for relevant information (Ambayo 2010). Academic digital libraries must play an important role not in controlling its contents but the services it provides. Its role must change from information provider to facilitator of information. Li and Chen, and Uppal and Chindwani noted that future libraries would concentrate to improve the utilization rate of library resources and to serve the users better (Li and Chen 2008; Uppal and Chindwani 2013). Regarding these issues, academic D-libraries needed to develop efficient systems that apply personalized services to better meet user's information needs (Stojanovski and Papic 2012; Huaxin and Qian 2013; Akbar, Shaffer et al. 2014). Personalized service is the process of presenting the right item/service to the right user (Speretta and Gauch 2005). The task of delivering personalized services is often framed in terms of a recommendation task in which the system recommends items and services to each user in a different way according to their preferences and needs (Smeaton and Callan 2005). The idea of the recommendation service is to help users in the effective identification of items suiting their preferences in a large space of possible options by predicting in advance their interest in an item/service (Porcel and Herrera-Viedma 2009; Kovacevic, Devedzic et al. 2010). Based on the users' individual preferences the goal of recommender systems is to find the best possible item for each user from a huge set of options (Mulvenna, Anand et al. 2000; Andonie, Russo et al. 2007). They are especially useful when they identify information that a user was previously unaware of (Tejeda-Lorente, Porcel et al. 2014). Because it's natural to reduce information overload (Lekakos and Giaglis 2006; Burke 2007; Akbar, Shaffer et al. 2014; Kun, Tingting et al. 2014), recommender system became a significant factor to increase the user satisfaction (Beel 2015). With regard to academic D-libraries, recommender system make it possible to offer academic D-library users potentially interesting resources (Morales-del-Castillo, Pedraza-Jiménez et al. 2009; Tejeda-Lorente, Bernabé-Moreno et al. 2014), and allows users to discover resources of interest outside their own keyword ideas (Li, Gu et al. 2009). Upadhyay mentions that recommendation service is an important academic D-library’s service to the users (Upadhyay 2015). The authors Li, Gu et al. added, more users are attracted to use digital libraries when recommendation service is included in the library management system (Li, Gu et al. 2009). There are two commonly techniques used in recommender systems based on how recommendations are made, which are collaborative filtering and content-based filtering. The collaborative filtering analyse a large amount of information on users’ preferences and determine recommendations to a target user based on their similarity to other users(Herlocker, Konstan et al. 2004; Bobadilla, Ortega et al. 2012). The collaborative filtering is considered one of the most used methods to generate personalized recommendations (Herlocker, Konstan et al. 2004; Lopes and Roy 2014). Research on collaborative filtering can be grouped into two methods: memory-based and model-based (Christidis and Mentzas 2013). Memory-based methods make similarity comparison across the entire user's historical database to find out the most similar users to the active user and then, recommendations are generated based on the similar users' rating. Different from memory-based methods, model-based methods require constructing a descriptive model of users using the user-item ratings and then recommendations are predicted using this descriptive model that can estimate the unknown ratings of a user. The other used approach in recommender systems is the content-based filtering, which generates the recommendations by comparing a set of keywords defined by the user with the items' features (Lops, de Gemmis et al. 2011).

It is natural that students prefer to get access to resources which are most influential on a particular topic with minimum time and navigation cost (Smeaton and Callan 2005; Porcel and Herrera-Viedma 2009). According to the study (Bhide, Heung et al. 2007) efficient management of large resources to provide an easy and quick access for students, is an important goal for D-libraries. Thus recommender systems based on collaborative filtering is a successful way to achieve that purpose. Amazon, iTunes, Netflix, and IMDB perhaps are the most well-known examples of collaborative filtering, when a customer selects a product the system presents a set of other products that the customer may be interested in. In academic digital library scope, user-based collaborative approach can be seen as a meeting place for students sharing common interests. Hence, such students might benefit from each other's knowledge by sharing experiences. The collaborative filtering approach learns from the experience of the first students to find useful resources for later students with similar information needs. There are two basic entities in the user-based collaborative filtering algorithm: User and Item. Users use rating to show their own opinion on items; this rating are represented byauser-item matrix (Symeonidis, Nanopoulos et al. 2008). The user-item matrix is employed for the similarity calculation to predict items to the active user based on a subset of users that are called neighbours who have similar tastes (Manh Cuong Pham 2011). For example, figure 1.1 shows a user-item matrix. The matrix has a dataset of m users and n items. Each user$_m$ has a set of items that they have rated in different scores between 1 and 5. Blanks indicate that the user$_m$ has not rated the item or the user$_m$ has not seen the item so far.

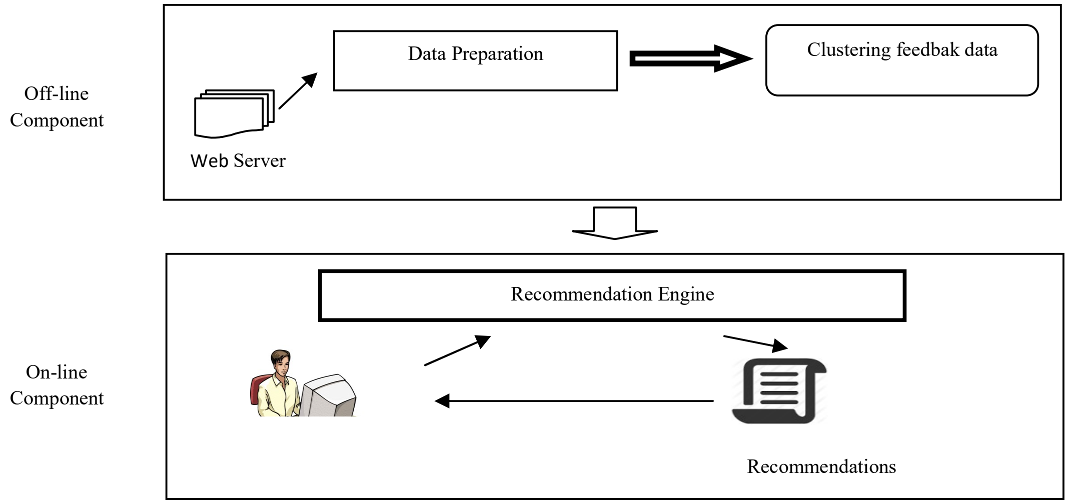
	Item$_1$	Item$_2$	Item$_3$	…	Item$_n$
User$_1$	3	5		…	
User$_2$	2		1	…	2
User$_3$	5	1		…	
…	…	…		…	
User$_m$	2		3	…	1

Fig. 1.1: A sample of user-item matrix

Academic D-libraries make its contents and services remotely accessible through the Web (Bundza 2009). D-library recommender systems based on Web provide recommendations by using two types of feedback data: explicit feedback data, e.g. ratings, and implicit feedback data, e.g. clicks. Most existing approaches used by web-based academic D-library, as well as approaches based on collaborative filtering, rely on explicit feedback to provide feedback on the resources and few studies addressed the issue of implicit feedback (Sahoo, Singh et al. 2012). Explicit feedback usually performs the feedback as a numeric rating scale or a binary like/dislike rating. Explicit feedback has the advantage of simplicity, recommender systems can easily use it, and it is often well understood by users. Nevertheless, explicit feedback suffers from a serious problem: it results in a static user profile, which is suitable for the recommendation process for some time after it is built; but its performance degrades over time as the profile ages (Choi and Geehyuk 2009). Also for a typical literature academic D-library, requiring students to rate resources makes the recommendation process not realistic because few students are willing to provide their interests resulting in a sparsity of evaluations explicitly. As the collaborative filtering process is based on computing similarity over the users to find similar users, the sparsity of evaluations prevents finding similarity between users, as there will not be an overlap between most users (Ahn, Kang et al. 2010; Wang, Yu et al. 2014). Even when user's neighbourhoods are defined, the computation of similarity between users is imprecise because of insufficient processed information, and consequently, it decreases the accuracy of collaborative filtering recommender systems (Ahn, Kang et al. 2010; Borhani-Fard 2013). This problem is referred as the "data sparsity" problem (Gong 2010; Lee 2015). As a result of "data sparsity" problem many item recommendations may be unable to make for users, this issue is known as reduced coverage (Su and Khoshgoftaar 2009; Kanimozhi 2011).

The aim of the study is to incorporate the clustering data mining technique and web usage data to alleviate the sparsity problem. Usage data provide information about user's activities when they enteraweb site until they leave it (Pallis, Angelis et al. 2007). These activities provide the system with implicit feedback data. Authors in (Lytras and Pablos 2011; Zhao and de Pablos 2011; Zhao, de Pablos et al. 2012) emphasizes the use of implicit rating techniques as an indicator of the user's interest for an item. In a Web-based academic D-library, after students perform their search, a list of results are provided by the D-library search engine. Each resource in the search results often is described by a number of link actions, i.e. the abstract link, the view link and the download link. Clicking on these links action provides so-called implicit user feedback, which allows the recommender system to discern the students' preferences. It is beneficial to consider a variety of observable student activities from these links action to infer students' preferences. When looking at these activities, a lot can be learned from it, e.g. if a student print a resource, the recommender system can assume that this resource is highly interesting for the student. Being able to make use of students' implicit data would have a number of advantages to D-libraries: a) provide the system with large quantities of data, b) achieves much greater coverage feedback judgments over resources than was achieved by using explicit data. In spite of the wealth of data provided by implicit data, recommender systems still suffer from the sparsity problem, especially when users access only a small portion of the available resources. Besides using implicit data, the study proposes to use clustering data mining to alleviate the sparsity problem. Clustering data mining is used to reduces the dimension of the student-resource matrix. The clustering also provides another benefit that is; the predictions for a particular student are generated from students sharing the same interests.

The study proposes a Web usage mining framework for academic D-library, the framework incorporates the clustering technique and Web usage data with the collaborative filtering approach. The Web usage mining framework consists of two components: off-line component and online component. The off-line component comprises two sequential stages: Data preparation and Data mining stages. In the first stage, usage data are transformed into numerical values. The data mining stage is responsible for deriving students clusters based on their rating using K-means clustering technique. The online component is responsible for providing recommendations to students based on student–resource matrix. The online component is composed of two sequential stages: constructing the student profile and generating recommendations. The first stage constructs the active student's profile (a student to whom recommendations are generated) on the basis of short-term interest. The student profile is built by observing the students' interaction with the resources in D-library Website. In the second stage, the student profile is compared to the clusters' profiles using the Cosine similarity measure to assign the active student to the best cluster profile. Then the Pearson correlation coefficient is used to compute the similarity between the active student and students within the chosen cluster to select the most similar students to the active student. Finally, a scoring method is used to select N-top resources for students. A generalized architecture of the proposed Web usage mining framework is depicted in figure 1.2.



The recommendation process starts when an active student provides the system with some feedback data. These feedback data is used to build a profile for the active student. The active student profile is then used by the recommendation engine to provide recommendations to the active student.

The rest of this chapter is organized as follows.Section 1.2 identifies the problem statement of the research followed by a brief description of the features of the proposed framework. Section 1.3 presents the motivation of the study. Section 1.4 describes the aim and objectives of the research. Section 1.5 presents the research question. Section 1.6 describes the research hypotheses. The importance and the contribution of the study are presented in section 1.7 and 1.8, respectively.

1.2 Problem identification

Academic digital libraries in universities include a huge number of resources and offer searching services in order to satisfy users' information needs. The huge amount of contents in academic D-library database present difficulty for library users to obtain needed information resources quickly and accurately (Jie, Haihong et al. 2012). Students face problems when acting with the academic D-library because they are often overwhelmed with a great number of resources, with the reality that students may not have the time or knowledge to evaluate these resources (Meyyappan, Foo et al. 2004). In addition, students are incapable of specifying precisely their needs in queries, especially when they are not familiar with the topic. They often find relevant information through trial or moving from one resource to another (Ambayo 2010).

This way is unproductive and leads to lower satisfaction among students due the time they spend on searching for suitable resources. Thus the number of students using digital libraries are reduced (Krishnamurthy and Balasubramani 2014). So academic digital libraries are facing challenges regarding their primary role of delivering relevant information to their users. In an effort to ensure that academic digital libraries achieve their objectives, academic D-libraries must move from being passive to being more proactive in offering information to users (Renda and Straccia 2005; Jange 2015). Tejeda-Lorente et al. mention that, information overload in D-libraries need easier access tools to help users to obtain relevant information (Tejeda-Lorente, Porcel et al. 2014). Proposed solutions to this challenge emphasize the need to build personalized recommendation system to support student's requirements in a simple and timely manner (Symeonidis, Nanopoulos et al. 2008; Tejeda-Lorente, Bernabé-Moreno et al. 2014). With personalized recommendation systems more students could turn their attention to libraries. On the other hand, the lack of personalized recommendation systems can make digital libraries underutilized by the students.

With the rapid expansion of information communication technologies in education, the Web became an integral part of today‘s educational practices. Accordingly, academic D-libraries make its contents and services remotely accessible through the Web. Many studies have been directed to build personalized recommendations for Web-based academic D-libraries. The most successful approach used by Web-based recommender systems is collaborative filtering technique. Collaborative filtering approach recommends resources based on the active user's preferences and the opinion of the users that are most similar to the active user. There are two approaches for students to provide their opinion for a resource, explicit feedback and implicit feedback. In explicit feedback approach, students are asked explicitly to rate resources. In implicit feedback approach, the interest on a resource is inferred from students' behaviour. Collaborative filtering recommender systems that rely on explicit feedback such as work by

Naak et al. (Naak, Hage et al. 2009) and Yang et al. (Yang, Wei et al. 2009) the user gives a numerical rating representing how much they liked the resource. However, it has been shown that collaborative filtering recommender system based on explicit feedback often results in less accurate recommendations than based on implicit data for several reasons: (a) Users may not be aware of their interests, (b) cannot deal with user preferences that change over time, which degrades the recommendation quality over time as the profile ages, and (c) there is a discrepancy between what the users report on evaluations and what they actually do. More importantly, it is a well-established fact that few students will spend additional time to provide rating (Brusilovsky, Farzan et al. 2005). When each student ignores rating resources or rates only a small number of resources, that results in a sparsity of evaluations, that is, there are a few of the total number of resources that are rated among available resources in the database. This problem referred to as the data sparsity problem, which has a negative effect on the efficiency of a collaborative filtering approach (Gong 2010; Lee 2015). As the collaborative filtering process is based on computing similarity over the users to find similar users, is difficult to define user's neighbourhoods due to the data sparsity problem because users will have little overlap in the resources they have rated (Ahn, Kang et al. 2010; Manh Cuong Pham 2011; Wang, Yu et al. 2014). Even when user's neighbourhoods are defined, the computation of similarity between students is imprecise because of insufficient processed information (Papagelis, Plexousakis et al. 2005; Su and Khoshgoftaar 2009; Borhani-Fard 2013), and consequently, it decreases the accuracy of collaborative filtering recommender systems. With the fast increasing of resources and users, the similarity is getting more difficult. The data sparsity also minimizes the coverage of the recommendation, this is known as reduced coverage (Su and Khoshgoftaar 2009; Chen, Wu et al. 2011; Kanimozhi 2011). Reduced coverage happens inthecase where there are only a few rated resources common among studentsome resources cannot be recommended at all. To clarify the problem, the following section shows how the sparsity problem impacts the D-library collaborative filtering recommendations quality.

The collaborative filtering recommendation process starts by finding the user's neighbours, i.e. similar users. For the purpose of finding the user's neighbours a matrix containing the users and their ratings on the items must be built (Gong 2010). In the D-library system, students' transactions indicate the interest on the resources consisting of students and resources and a value that represents the degree of interest for the resources accessed by students. When multiple students' transactions are concerned, the students and their rating on resources can be represented in a matrix. This matrix is called the student–resource rating matrix. Figure 1.3 shows the student–resource rating matrix contains a set of M students and N resources, where R_{mn} denotes the score of the resource n rated by a student m. If student m has not rated resource n, then R_{mn} =?. Such type of rating matrix is difficult to define user's neighbourhoods because users have little overlap in the resources they have rated.

	Resource$_1$	Resource$_2$	Resource$_3$	………	Resource$_n$
Student$_1$	?	R_{12}	?		R_{1n}
Student$_2$	?	R_{22}	R_{23}		?
Student$_3$	R_{31}	?	?		R_{3n}
Student$_4$	?	?	R_{43}		?
…….	…….	…….	…….	…….	…….
Student$_m$	R_{m1}	?	R_{m4}		R_{mn}

Fig. 1.3: Student-resource matrix

The following example shows asparsity user–resource rating matrix for five students and eight resources in a resource recommender system.

Example 1.1: In the example, we will go through the basic ideas of matrix approach. Given the m × n student–resource rating matrix includes 5X8 = 40 elements such that there are 5 students S = { St_1, St_2, ..., St_5} and 8 resources R = { R_1, R_2, ... , R_8}. A real D-library would have thousands of users and resources, but for illustrating the problem, the example uses a smaller set.

The student–resource rating matrix rows represent the students, and the columns represent the resources. The row r_i represents the interests of student i and consists of a set of resources which indicates the student's interest in those resources. The student interest in the resources is assigned by a value F where F is the set of student feedbacks, F = {1, 2, 3, 4, 5}. Otherwise, the student interest is 0, 0 indicate that the student has not rated a particularresource.

$$S = \left\{ \begin{array}{l} s \in F \text{ if student i rated resource j} \\ s = 0 \text{ Otherwise} \end{array} \right\}$$

We have 5 students and 8 resources, each student i has a set of items that they have rated them in different integer scores ranging from 1 to 5, the matrix may look as shown in figure 1.4. From the first line in the matrix, the student St_1 rated the resources R_2 and R_4 andR_5, the rating is 3 and 2 and 4 respectively. From the second line in the matrix, the student St_2 rated the resources R_5 and R_7; the rating is 4 and 5 respectively. From the third line in the matrix, the student St_3 rated the resources R_4, R_5, R_7, and R_8, the rating is 2, 2, 3, and 4 respectively. From the fourth line in the matrix, the student St_4 rated the resources R_2 and R_4 and R_8; the rating is 4 and 5 and 5 respectively. From the fifth line in the matrix, the student St_5 rated the resources R_4, R_5, R_7, and R_8 the rating is 3, 2, 4, and 3 respectively. As shown in Figure 1.4 most student-resource pairs have blanks, meaning that the students have rated a few resources among the total number of available resources.

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
St_1	0	3	0	2	4	0	0	0
St_2	0	0	0	0	4	0	5	0
St_3	0	0	0	2	2	0	3	4
St_4	0	4	0	5	0	0	0	5
St_5	0	0	0	3	2	0	4	3
Active St_1	3		2			4		
Active St_2		2		4				2

Fig. 1.4: A utility matrix of resources

Given the student-resource rating matrix, we would like to know which resources will be of high interest to the student “Active St$_1$” and student “Active St$_2$”, such that we can make recommendations to them. The collaborative filtering recommender system recommends resources based on the active user’s preferences in the current session and the opinion of the students that are most similar to the active student.

In describing the use of user-based collaborative filtering approach, three phases are followed for recommending resources for an active student:

- Calculating the similarities between active student rating and previous students’ rating.
- Selecting nearest students to the active student, known as active user’s neighbours, who will serve as recommenders.
- Use the ratings from active student’s neighbours to calculate prediction score for the resources.

For the purpose of calculating the similarities between active students and other students Cosine distance metric (equation 1.1) is used in this example. For the purpose of selecting the most similar students to the active students, threshold-based selection technique is used in this example. Threshold-based selection technique select users whose similarity exceeds a certain threshold value.

$$Cosine(x,y) = \frac{\sum_{k=1}^{n} \mathrm{R_{xk}}\,\mathrm{R_{yk}}}{\sqrt{\sum_{k=1}^{n} \mathrm{R_{xk}}^{2}}\sqrt{\sum_{k=1}^{n} \mathrm{R_{yk}}^{2}}} \qquad \text{Equation (1.1)}$$

In equation (1.1) R_{xk} indicates the rating of the resource k by student x, R_{yk} is the rating of the resource k by student y and n is the number of items co-rated by both users.

Recommendations for the student "Active St$_1$":

From the student-resource matrix we can see that the student "Active St$_1$" does not share any rating patterns with any of the students in the matrix. Consequently the system is unable to recommend resources to the student "Active St$_1$".

Recommendations for the student "Active St$_2$":

Step 1:

The cosine of the angle between student St$_1$ and student "Active St$_2$" is computed as follows:

$$\frac{2x3 + 2x4}{\sqrt{3^2 + 2^2 + 4^2}\sqrt{2^2 + 4^2 + 2^2}} = \frac{14}{26.381} = 0.53$$

The cosine of the angle between student St$_2$ and student "Active St$_2$" is computed as follows:

$$\frac{0}{\sqrt{4^2 + 5^2}\sqrt{2^2 + 4^2 + 2^2}} = 0$$

The cosine of the angle between student St$_3$ and student "Active St$_2$" is computed as follows:

$$\frac{2x4 + 4x2}{\sqrt{2^2 + 2^2 + 3^2 + 4^2}\sqrt{2^2 + 4^2 + 2^2}} = \frac{16}{28.142} = 0.568$$

The cosine of the angle between student St$_4$ and student "Active St$_2$" is computed as follows:

$$\frac{4x2 + 5x4 + 5x2}{\sqrt{4^2 + 5^2 + 5^2}\sqrt{2^2 + 4^2 + 2^2}} = \frac{38}{39.790} = 0.955$$

The cosine of the angle between student St$_5$ and student "Active St$_2$" is computed as follows:

$$\frac{3x4 + 3x2}{\sqrt{3^2 + 2^2 + 4^2 + 3^2}\sqrt{2^2 + 4^2 + 2^2}} = \frac{18}{30.190} = 0.59$$

In the Cosine measure metric a larger (positive) cosine outcome implies a smaller angle,and therefore a smaller distance, accordingly active student Active St_2 is closer to the students St_4 and St_5 than St_1, $St_{2,}$ and St_3.

Step2:

For selecting the most similar students a threshold value is defined. With threshold value, the number of active student's neighbours who will serve as recommenders can be controlled. If it is low, many neighbours will participate by their resources' rating for predicting recommendations, resulting in low precision and high coverage of resources. If it is high, few neighbours will participate by their resources' rating for predicting recommendations, resulting high precision and low coverage of resources (Singh and Singh 2010; Manh Cuong Pham 2011). From the example above the sparsity problem provides the recommendation system with a few students due to low similarity.

As can be seen from figure 1.4 the vast majority of ratings are unknown, however, real data in D-library are more sparsity than it is in the example presented above. In real D-library the rating matrix will increase along with the growing of the users and resources. Thus the matrix will be very high dimensional. With high-dimensional matrix, it is difficult to extract common interest users (Wang, Yu et al. 2014). Both the sparse data sets and the high dimensional space in D-library lower the accuracy of the recommendations.

The weakness of D-library recommender systems based on collaborative filtering related to using explicit feedback led us to explore solutions for the problem. The research problem is related to the incorporate of usage data with clustering technique based collaborative filtering to alleviate the sparsity problem. On this basis, a Web recommender framework based on implicit feedback and K-means clustering technique is proposed whose main features are the following:

- Dividing the student-resource matrix: K-means clustering technique is used to divide the sparse student-resource matrix into partitions. By this way, the sub-matrices become increasingly “dense” as the dimensionality decrease.
- Student preferences captured at a finer granularity: The study considers the granularity of interaction on the basis of detailed observations of students’ interaction to provide additional information for alleviating the sparsity problem. The granularity of interaction data can reduce the sparsity problem because there is a large amount of continuous flow of such data. The granularity of interaction sheds light on the potential usefulness of a resource. This framework monitors five common user’s actions that can occur after the student follows a link from the search results: printing a resource, bookmarking a resource, downloading a resource, reading a resource, viewing a resource abstract.
- Student similarity measures: In the proposed framework,clustering technique and K Nearest Neighbours (KNN) are used in a cascade. Based on the collaborative filtering, after clustering the student-resource matrix the framework needs to derive the similarity function, the study proposes two similarity metrics to derive the similarity function; the first metric is the Cosine similarity metric which is used for matching the active student with the clusters’ profile in order to choose the best cluster. The second similarity metric is the Pearson correlation coefficient, which is used to select the most similar users to the active user.
- Weighting actions method: The framework includes a method to model the student’s interest on a resource through their activities. This includes five actions: printing a resource, bookmarking a resource, downloading a resource, reading a resource and viewing a resource abstract. These actions provide different levels of evidence for interests. As a result of that, a weighting method for these actions is required. The weighting method weight actions according to their level of evidence of interest, where

actions with a higher level of evidence will have higher scores than actions with alow level of evidence.

- Building short-term interest model: Usually, recent feedback is more relevant than the older one. To take this into account, the proposed system builds a short-term interest model to represent the student's interests. Adapting the recommendation strategy to the current student’s short-term interests helps to recommend more accurately what the student will finally access. Gauch mentioned that building short-term interest model for the active user leads to a user model that can adjust more rapidly to the users’ changing interests (Gauch, Speretta et al. 2007).
- Rank the recommendation results: A recommender system should provide a ranking list to the users to minimize the effort required to find high quality resources. From the recommendation perspective, the order over the items is an important issue (Pessiot, Truong et al. 2007).

1.3 Motivation

Academic D-library is becoming an increasingly important issue in the world of higher education because it present asophisticated and reliable way for acquiring knowledge to students (Linghui 2010; Anaraki and Heidari 2011). Searching for relevant resources in the academic D-library is an essential service in higher education institutions. Mayega mentions that recently libraries management systems give more respect to providing users’ requests than the resource they collect (Mayega 2008).

As academic digital libraries contents become more varied and huge, it is difficult for students to obtain the needed information resources quickly and accurately. Thus, students expect more sophisticated services from digital library systems such as easy to retrieve resources

(Tejeda-Lorente, Porcel et al. 2014). Due to the generality of the terms used by students during a search, and the lack of information about resources, it is hard for students to find relevant resources, then, not surprisingly students will ignore these results (Tejeda-Lorente, Porcel et al. 2011). One effective solution to handle this issue is to make use of the personalized recommendation service that provides each user with a list of resources that they would be interested in. Personalization of academic D-library is considered one of the most important tasks in the area of digital libraries (Furner 2002; Tejeda-Lorente, Porcel et al. 2014). One of the important personalization services is the recommendation service whose objective is to evaluate and filter the huge amount of resources to assist the users in their selection. The overwhelming number of resources available in academic D-library and the vast amount of resources added periodically, and most students are incapable of specifying precisely their needs make the academic D-library domain a good target for recommender systems. Upadhyay mentioned that recommending resources is considered as an important task for academic D-library users. Also, he mentioned that supporting the users' services in the D-library to meet out their expectations is going to be the primary challenge in coming years (Upadhyay 2015). On the other hand, lack of recommendation service can make academic D-library inconvenience for students.

Many academic D-libraries make its contents and services remotely accessible through Web to allow more users to access its resources easily (Diallo and Liwen 2011). These systems usually use collaborative filtering to recommend resources to users. Typically, Web-based collaborative recommender systems need to get preference data from users in order to provide recommendations (Jawaheer, Weller et al. 2014). Existing approaches used by web-based academic D-libraries, as well as approaches based on collaborative filtering rely on obtaining preferences by two ways: explicit rating and implicit rating. Explicit ratings are preferences provided explicitly by users, and implicit ratings are inferred from user behaviour as they interact with the system (Tejeda-Lorente, Bernabé-Moreno et al. 2014). Although the explicit rate obviously indicates what students believe is useful and relevant (Brusilovsky, Farzan et al. 2005), it has some drawbacks for D-library recommender system which are: (a) students may not be aware of their interests and (b) requiring students to rate the selected resources is not realistic. Accurate recommendations undoubtedly depend upon the degree of which the recommender system has incorporated the relevant information about users' preferences into the recommendation engine.

The recommender system includes agroup of users and a set of resources. Users have rated some resources in the system. Users and their rates on the resources are represented in a matrix. Recommender systems use the matrix to recommend items to target users by matching their preferences against the entire user-item matrix to find a set of users known as neighbours. Items that the neighbours prefer are then recommended to the target user. Using explicit rating provide insufficient ratings for the user-item matrix. This problem called “data sparsity” problem has a strong effect on the accuracy of collaborative filtering recommendation systems (Ahn, Kang et al. 2010; Dakhel and Mahdavi 2011; Sushmitha, Annushya et al. 2015). In D-library recommender system the data sparsity problem occurs when each user rate only a small number of resources. Since the correlation is only defined for students who have rated at least two resources in common, many active students will not have acorrelation at all. Hence the system will be unable to make many recommendations for a particular student. Also, there will be resources which never be recommended to students. This problem is known as reduced coverage problem (Su and Khoshgoftaar 2009).

The motivation for this work is based on the use of implicit feedback with clustering mining for D-library collaborative filtering system to alleviate the sparsity problem and to provide more accurate predictions for D-library recommender systems.

1.4 Aim and Objectives of Research

On-line academic libraries became resource centres for education and research (Stojanovski and Papic 2012). Nowadays, academic libraries offer a huge amount of information resources, such as electronic journals, electronic books, and electronic papers. The rapid growth of these information resources overwhelm users and makes it difficult for them to access to relevant information. Tejeda-Lorente mentions that students need easier access to the thousands of resources to find relevant and useful information (Tejeda-Lorente, Porcel et al. 2014). Analysing large volume of student's access data poses opportunities for the academic D-library to identify students' preferences. This necessitates the development of methods for analysing such data. The aim of this study is to present a Web usage framework based on clustering technique that uses students' feedback data with collaborative filtering to improve academic D-library recommendation system. To achieve the proposed aim, the objectives of this study are accordingly stated as follows:

1. To develop a collaborative filtering recommender framework that utilizes clustering mining.

The collaborative filtering process is based on computing similarity over the users to find similar users. Nearest neighbour algorithms rely upon exact matches between users may be unable to make recommendations for many users,and many resources may not be recommended forever. This problem is known as sparsity problem and is due to large levels of sparse data sets, that will lower the accuracy of the recommendations. Moreover, along with students and resources increase, the sparsity problem will increase. In this work, we model the students' feedback data using K-means clustering technique so as to be exploited online during the target student's session.

2. To leverage the implicit feedback data to make accurate decisions for the recommendations.

Implicit recommender systems rely on the observation of user activities. Thus student's usage data (access data) can act as a very rich source for the proposed collaborative filtering framework. Student actions provide so-called implicit user feedback these actions provide different levels of evidence for interest for inferring student’s preferences. Using these implicit data will lead to increase the number of ratings, subsequently enhance the recommendations accuracy. In addition, collecting feedback data from any action done by a student will help to cover a wider range of resources to be recommended.

1.5 Research Question

With the e-business development in education, Web-based academic D-library becomes more and more important for higher education. Collaborative filtering recommender systems compare active user feedback data to other users to find users whose past feedback data is similar to that of the active user and use their feedback data to predict what the active user would like. The sparsity problem makes the computation of similarity between users imprecise, and consequently reduces the accuracy of the recommendations. The purpose of this research is to combine the Web usage pattern knowledge into the collaborative recommendation service to alleviate the sparsity problem. By reducing the sparsity problem, the quality of the recommender system should improve. Student's feedback data based on the access behaviour provide the recommender system with more data. Based on the above discussion, the primary research question for the study is: "Does Web usage mining used by Web collaboration recommendation approach provides accurate recommendations to academic D-libraries students when there are too much sparse data sets?”. To develop the proposed frameworktwo more specific sub-questions follow the general research question:

Research question 1 How does clustering feedback data help alleviates the sparsity problem?

In order to come up with accurate recommendations, the sparsity problem must be alleviated. We discuss how to alleviate the sparsity problem by dividing the student-resource matrix into several low-dimensional dense student-resource matrices, then use these matrices instead of the original matrix to provide recommendations.

Research question 2 Can the use of implicit feedback data help alleviate the sparsity problem?

In order to come up with accurate recommendations, the system has to deal with plenty of user feedback data. Therefore, there is a need to show how using implicit feedback data can have a positive effect on the sparsity problem. Five different actions for implicit feedback are proposed, including printing, bookmarking, downloading, reading and viewing a resource abstract. Using this implicit feedback data will provide the recommender system with plenty of data helping the recommender system to alleviate the sparsity problem,subsequently find similarity between users.

1.6 Research Hypotheses

As the number of the academic D-library contents are getting larger and larger in recent years, the study recommends relevant resources from a large amount of resources to students. Using the implicit feedback data with the explicit feedback data as a collaborative information source will yield more effective recommendations. The following hypotheses are, as a result, formulated:

H1: Using Web clustering mining technology significantly produces effective recommendation.

In real D-library the rating matrix will increase along with the growing of the students and resources. Thus the matrix will be very high dimensional. With high-dimensional matrix the sparsity level will increase, which make it difficult to extract common interest. The sparse rating matrix has a negative effect on all collaborative filtering methods performance. Clustering data mining technique is used to divide the student-resource matrix into several low-dimensional dense student-resource matrices, then these matrices are used instead of the original matrix to provide recommendations, thereby can gain effective recommendations. To check this, the Mean Absolute Error (MAE) metric and Sparsity level equation are used. The Mean Absolute Error (MAE) metric is examined for two experiments: The first experiment evaluates the quality of recommendations generated by a memory-based collaborative filtering method, and the second experiment evaluates the quality of recommendations generated by using k-means clustering technique. The Sparsity level equation is used to measure the sparsity level of the original matrix – before clustering-, and for the sub-matrices after applying the clustering approach.

H2: Recommendation approach that considers the implicit feedback data will result in more accurate recommendations.

The system must be able to assist students in the selection of resources through the information related to student's activities on the D-library Website. Usage data provide a detailed record of every single action taken by the user during interacting with D-library Website. Thus, it provides the recommender system with plenty of feedback data. Producing recommendations from a rich source of feedback data have a positive effect on the recommendations accuracy. To check this, the Mean Absolute Error (MAE) metric is examined throughout using only explicit feedback data compared to using explicit and implicit feedback data.

1.7 Importance of Study

In support of the University's mission, the academic digital libraries support students to access and use information for academic success (Stojanovski and Papic 2012; Adeniran 2013; Soria, Fransen et al. 2013). Therefore, expanding libraries' e-services become necessary to stay leading educational institutions (Stojanovski and Papic 2012; Papic and Primorac 2014). Stojanovski and Papic noted that educational services became the most growing segment of digital library services in recent years (Stojanovski and Papic 2012). Therefore, library management systems must provide sophisticated services for students to make adequate use of the learning resources.

Academic digital libraries contents a huge number of resources, the number of users accessing it are growing at a tremendous rate. Thus, libraries must develop new approaches for easy access to relevant information resources (Jie, Haihong et al. 2012). Traditional digital libraries allow students to query certain keywords, and return search results purely based on keywords input. In this scenario, a student could hardly retrieve resources outside of their own search keywords. According to Meghabghab and Kandel, and Tejeda-Lorente et al. many times when the students try to receive useful articles, they obtain irrelevant articles (Meghabghab and Kandel 2008; Tejeda-Lorente, Porcel et al. 2014). Therefore, focusing on their expectations is an important issue. A service that provides users' expectations in academic D-libraries is the recommendation service, which provides relevant resources to library users. Recommendation service is a service used to evaluate and filter the huge amount of information available on information systems to assist users in their information access processes (Kveton and Berkovsky 2015). With the recommendation service, users are presented with a list of recommended resources based on their interest. Consequently, the utilization ratio of digital resources is increased and more students could turn their attention to libraries. Pinata mentions that the recommendation service in the educational libraries is an important issue (Winoto, Tang et al.

2012). In the realization of the academic digital library, recommendation service plays an important role; it has the effect of guiding the student in a personalized way to relevant resources in a large space of possible options in a short time. Upadhyay mentions that academic D-libraries are seeking to provide recommendation service as it is a valuable library's service to the users (Upadhyay 2015). The authors Li, Gu et al. added, more users are attracted to use digital libraries when recommendation service included in the library management system (Li, Gu et al. 2009).

Academic D-library system must understand the users and have to provide the services accordingly to accomplish adequate personal recommendation service (Renda and Straccia 2005; Dianjun and Min 2011; Krishnamurthy and Balasubramani 2014). As academic D-libraries eventually go online, there is a possibility of suggesting resources to students based on their implicit feedback data. Pohl et al. mention that, recommender systems based on implicit feedback data generate interesting recommendations (Pohl, Radlinski et al. 2007). Usage data is a perfect source to provide students' interest because it continuously records all students' activities while browsing the D-library Website. Upadhyay noted that information behaviour of users is a critical factor that D-libraries must consider (Upadhyay 2015). Thus, it is beneficial to consider a variety of observable student activities to discern the student's feedback about the search. User actions like reading a resource or downloading a resource provide so-called implicit user feedback which allows the recommender system to infer students' preferences. These actions derive students' preferences when accessing resources each action provides a different level of evidence of interest. Usually, a student is unlikely to click on a resource they consider less relevant than another resource they observed. From this logic, it is reasonable for academic D-library to incorporate usage data into their recommendation engines. This study presents a collaborative filtering framework based on the behavioural patterns of students. This is called implicit recommendation service which relies on the observation of user activities. The proposed framework analyses implicit relevance feedback extracted from observed student behaviour, in particular, click data provided by log files. Click data is more reliable than other forms of implicit rating (Joachims, Granka et al. 2005) and it is easy to collect (Joachims 2002), and it.

This study has great importance for Web recommendation service provided by academic D-Library for the following reasons:

- The study contributes to the achievement of the goals of online academic D-libraries. The proposed framework makes digital resources more accessible through advising students with the best resources; subsequently, students could easily find resources they are not aware of. This will increase the library resources utilization ratio since regular students might use more resources and more students are attracted to use the D-library.
- Collaboration Web recommendation for D-library represents an educational support innovation that significantly can contribute to the high educational system. The proposed framework discovers collaboration possibilities based on similarity allowing students to take advantage of past searches performed by similar users. Such task presents the digital library as a collaborative environment where students can share interests with other students.
- Model-based collaborative filtering methods can produce more precision recommendations and achieve good online performance in comparison with memory-based collaborative filtering methods (Kuzelewska 2014; Sushmitha, Annushya et al. 2015). Thus, the use of web usage clustering mining technique is important for Web-based D-library since the clustering technique can find out groups of students based on their similarity. These groups contain students that are more similar to each other than those in other clusters. This allows avoiding the participation of far studentsin generating recommendations, consequently yielding higher recommendations accuracy.

- One principal disadvantage of the collaborative filtering approach is the need for a lot of ratings to obtain a good performance. The sparse rating matrix has a negative effect on all collaborative filtering methods performance. In contrast, in case of dense rating matrix, all methods tend to achieve higher performance (Hernando, Jes et al. 2016). This study offer clustering technique incorporated with implicit feedback data for performing the density of the student-resources matrix.
- The feedback involves granularity of interaction data into the feedback. Involving granularity of interaction data can efficiently improve the user feedback. The studies (Cooley, Mobasher et al. 1997; Srivastava, Cooley et al. 2000) noted that analysis of the user's browsing patterns could provide recommendations according to the user preferences. The proposed framework considers various user actions from usage data which provide an in-depth understanding of the behaviour of students and produce more comprehensive representation for student profile.

1.8 Contribution

The key contribution of this study is that we can successfully use usage data for the recommendation process and demonstrate the value of mining data to extract interesting patterns for detecting active student's needs. Based on this approach, a Web usage mining framework collaborative filtering that incorporates the clustering technique and usage data is designed. The efforts in this work are summarized in the following contributions:

- **Using implicit web usage data to infer the user's preferences.** To recommend items to the online user, it's essential to understand the user interactions with the system (Jawaheer, Weller et al. 2014). Wang and Ren mentioned that great knowledge could be extracted from the system-user interaction (or usage data) (Wang and Ren 2009).

Adomavicius and Tuzhilin (Adomavicius and Tuzhilin 2005) argued that future recommendation system research has to rely more on implicit feedback. Using implicit feedback for collaborative filtering provides many benefits: a) Provide data without any additional burden on the users (Rendle, Freudenthaler et al. 2009), b) it is immediately available (Pohl 2006; Jawaheer, Weller et al. 2014), c) provides better coverage than explicit data, d) more reliable than explicit feedback as users provide their interest without they are aware (Schafer, Frankowski et al. 2007), e.g. In case of using explicit feedback not providing feedback by a user can mean that the user did not see the resource or they didn't like it, or they liked it but were not interested in providing feedback for it. Thus, in such circumstances, recommendation algorithms should consider usage data as a reliable data for the decision-making process.

Usage data available in Web-based academic D-library represent a potential gold-mine for libraries to identify the online students' interests in term of implicit feedback. Potentially, every student interact with the D-library Website will generate implicit data reflecting their preferences. This implicit feedback data provide the system with plenty of data rather than the sparse data encountered by explicit user feedback. Different actions are available on D-libraries, in this study, five actions are identifiedinclude "printing", "bookmarking", "downloading", "reading" and "viewing abstract", each can be used as an indicator of the student preferences.

Since collaborative filtering needs numerical values to explore similarity between students, in the case of implicit user feedback the system must convert the implicit feedback into a numerical value (Lichtenstein and Slovic 2006; Koren 2010). Many studies made a correlation between implicit and explicit feedback, e.g. reading time (Konstan, Miller et al. 1997), printing an article (Oard and Kim 2001), music playcounts (Denis Parra 2011). This study represents a numerical value of the five actions. These actions provide different levels of evidence for interests. Based on this concept printing a resource produces five points, bookmarking a resource produces four points, download a resource produces three points, reading a resource produces two points and ignoring a resource after viewing its abstract produces one point. Ignoring a resource is assumed to be of no interest and is given a Zero in the rating score.

- **Clustering student-resource interaction matrix**. Model-based collaborative filtering algorithms handle the sparsity better than memory-based collaborative filtering algorithms. Along with students and resources increase, the student-resource matrix will be very high dimensional. A k-means clustering model is proposed to alleviate the sparsity problem because it reduces the dimension of the student-resource matrix, this transforms the original student-resource matrix into a lower dimensional space. Reducing the dimensionality result a less sparse matrix than its high dimensional, thereby providing more accurate recommendations.
- **Using negative implicit feedback.** In recommender systems based explicit feedback, it is easy for users to tell the system what they found unsuitable to their needs (Chao, Balthrop et al. 2005). In contrast, it is hard to acquire negative feedback through implicit feedback (Gauch, Speretta et al. 2007). Negative feedback is helpful to distinguish between relevant and non-relevant resources which significantly improve recommendation quality (Hu, Koren et al. 2008; Lee and Brusilovsky 2009; Peska and Vojtas 2013). Thus, it is crucial to add negative feedback. None of the previous studies in the D-library recommender systems utilized negative preference in implicit feedback. The five actions proposed in this study cannot indicate a negative preference. For example, a student that did not download or read a resource might have done so just because they did not know about the resource. However, in this study, a simple mechanism is introduced to infer negative preference from implicit feedback by

aggregating the action “viewing abstract” with the other actions. The action “viewing abstract” is used primarily by D-libraries for checking the resource whether it is suitable or not. When students click the “abstract” option they can see an overview of the resource, this is considereda “checking” point, if they find the resource interesting, they will print it or at least read it. Otherwise, they close the abstract window and search for other resources, in this case, we can assume that the resource did not match their interests. The proposed framework considers viewing a resource abstract without performing any of the rest actions as a negative preference.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter involves six sections. Section 2.2 reviews background knowledge that is important for understanding the context of the problem and the techniques that will be used throughout this study. This includes a review of the digital library, academic digital library, recommender systems, Data mining, and Web mining. Section 2.3 describes the relationship between data mining and digital library, as well as the applications of data mining in the digital library. In section 2.4, the existing methods and techniques used to alleviate the sparsity problem are discussed. Section 2.5 reviews related studies to the collaborative approach and explain how it relates to the proposed work. The summary of this chapter is given in section 2.6.

2.2 Background

This section is divided into five sub-sections. Section 2.2.1 defines digital library. Section 2.2.2 defines academic digital library. Section 2.2.3 defines web personalization recommendation systems and explain how recommendations are generated. Section 2.2.4 provides an overview of Data mining. Section 2.2.5 provides an overview of Web mining.

2.2.1 Digital Library

Libraries are considered as the earliest form of knowledge for readers. The library's traditional role is a repository for physical collections of books and journals (Board 2011). Due to the advancement of the information and communication technology, libraries adopted the technology to transfer from physical to digital libraries (Adeleke and Olorunsola 2010; Dianjun and Min 2011; Kadir, Rahman et al. 2014). Digital libraries are the logical extensions of physical libraries in the information age (Kadir, Rahman et al. 2014). Although the traditional objectives of the library remain primarily the same, the method of executing and delivering information has changed, and the format of information resources is changed to electronic form due to the impact of information technology (Odongo 2011). A study by (Keralapura 2009) found that the use of information technology in libraries has a positive impact on user satisfaction.

A digital library system (also known as an E-library) simply is an information resource system supported by many technologies to store a large resources in digital format, and organize and manage it, and provide access to users through the network (Buchanan, Bainbridge et al. 2005; Tejeda-Lorente, Porcel et al. 2014). Digital Libraries are making perfect use of digital capacity to provide services such as organizing (Lei and Lingshui 2010), information seeking (Mohd Shuib, Abdullah et al. 2010), and delivering (Aghakhani, Najar et al. 2010). The digital library has different definitions in the literature, according to (Spink, Wilson et al. 2002) digital library is a set of integrated services for storing, cataloguing, filtering, seeking, and retrieving information. Digital library resources can be scientific, business or personal data, and can be represented as digital text, image, audio, video, or other media.

The digital library helps readers to quickly and effectively find the information they need without necessarily having the need to be physically present in a traditional library building and without time restriction (Rahman 2008; Zhang 2011; Adeniran 2013). The well-designed digital library system has the potential to enable non-specialist users to query efficiently and easily retrieve information (Ogunsola 2011).

The following points present the advantage of the digital library system over the traditional physical library (Brangier, Dinet et al. 2009; Porcel, Moreno et al. 2009; Abdullah and Kassim 2012; Stojanovski and Papic 2012):

- No physical space requirements.
- Contains different digital content types, e.g. documents, images, maps, audio, and video.
- Easy to add new resources to a digital library.
- Easy to update the digital resources from the primary sources.
- The possibility of remotely accessible through computer networks which allow access to broader users.
- Elimination of concern for library resources being lost or damaged.
- Easy to archive the digital library content.
- Easier accessibility to resources that can be searched manually.

2.2.2 Academic D-Library

Academic digital library, like any other digital library, is an information resource system. Academic digital libraries represent digital libraries of higher education organizations which provide high-quality learning resources and services to faculty, researchers and students (Bazillion 2001; Dollah, Ab et al. 2006; Kadir, Dollah et al. 2009). Academic digital libraries present a sophisticated, faster, simpler and reliable tool to acquire knowledge in education organizations (Razilan, A.K. et al. 2009).

In (Dollah 2008) the importance of the role of academic digital libraries in the dissemination of knowledge is emphasized to increase the students' knowledge in academic institutions. Adeniran mentions that the use of electronic resources in the libraries is necessary for universities development (Adeniran 2013). There are benefits that academic digital library can provide in education (Han and Goulding 2003; Smith 2005; Nichols, Bainbridge et al. 2006; Ross and Sennyey 2008; Board 2011; Palmer 2012; Adeniran 2013):

- Curriculum planning (Carlson and Reidy 2004; Maull, Saldivar et al. 2010).
- Designing teachers' courses (course development) (Barker 2009).
- Creation of an environment which contribute to faculty and students research (Recker, Walker et al. 2007).
- Provide an easy tool for students to find information relevant to their courses (Adeniran 2013).
- Help in distance learning and e-learning university programs (Sharifabadi 2006).
- Teachers can share resources in ways that are not practical with paper-based resources (Impagliazzo, Lee et al. 2003).

Resource on the Internet can be used in the academic digital library, in this case, the resources will be presented in a controlled environment which will provide the following benefit for students:

- The resources focus on student activity. Students learn better when resources are relevant to their own situation.
- Limit access to some resources for reasons of student academic level.

2.2.3 Web Recommendation System

2.2.3.1 Introduction

A recommender system is a software which helps users in finding items in a large space of possible options suiting their preferences (Mönnich and Spiering 2008), these users don't have a detailed item domain knowledge (Ricci, Rokach et al. 2011). It is used in information systems as a means to help users cope with the information overloading (Kveton and Berkovsky 2015). According to (Liang 2008; Malinowski, Weitzel et al. 2008; Esteban, Tejeda-Lorente et al. 2014) a recommender system could be seen as a decision support system, where the solution alternatives are the items to be recommended and the criteria to satisfy are the user preferences. Such systems have become powerful tools in many domains, such as, e-commerce (Castro-Schez, Miguel et al. 2011), social network (Guy, Zwerdling et al. 2010), education area (Bobadilla, Serradilla et al. 2009; Konstan, Walker et al. 2014), tourism (Bobadilla, Serradilla et al. 2009; Fuchs and Zanker 2012; Xuesong and Kaifan 2013), digital library (Tejeda-Lorente, Bernabé-Moreno et al. 2014), Web search(McNally, O et al. 2011). The recommended items may include books (Crespo, Martínez et al. 2011), news (Lin, Xie et al. 2014), images (Yang, Wang et al. 2014), films (Koren, Bell et al. 2009), TV show (Koren, Bell et al. 2009), music (Lee, Cho et al. 2010).

Since the provision of recommender systems requires a thorough knowledge of users' preferences (Tejeda-Lorente, Bernabé-Moreno et al. 2014). This knowledge implies that the system presents actual user interest in the form of a profile (Ambayo 2010). The authors emphasized that successful recommendations mainly depend on accurate users' profiles (Quiroga and Mostafa 2002; Porcel, Moreno et al. 2009; Tejeda-Lorente, Porcel et al. 2014). The user profile can be built in different ways: Some can be build by utilizing personal preference information, such as services or items in which the user is interested. This type of profiles is called the user interest profile (Adomavicius and Tuzhilin 2005; Shani and Gunawardana 2009). Some can be build based on demographic information (Chen and Chen 2007). In principle, there are two types of the user interest profile, profiles based on explicit feedback and profiles based on implicit feedback. Many systems adopt a hybrid approach (Hu, Koren et al. 2008; Parsons, Ralph et al. 2011). Explicit profiles are obtained from information provided by users when they are asked to evaluate services or items (Li, Wang et al. 2010). This approach is the most common (Porcel, Castillo et al. 2010). Examples of data collection for building explicit profiles include the following (Burke 2002; Schafer, Frankowski et al. 2007):

- Asking a user to rate an item scale, like numbers ranging from 1-5.
- Asking a user to rank a collection of items, from favourite to least favourite.
- Asking a user to choose the better, one of two items.
- Asking a user to create a list of items they like.

Implicit user profiles are automatically extracted by observing user behaviour over time as they interact with the system (Gauch, Speretta et al. 2007; Lops, de Gemmis et al. 2011). The system updates the user's profile by detecting changes while observing the user (Porcel, Castillo et al. 2010). Examples of data collection for building implicit profiles include the following:

- Analysing purchase history.
- Observing mouse activities, scrollbar activities, keyboard activities, and time spent on the Website page.
- Obtaining a list of services that a user has used.
- Analysing the user's social network.

There are many recommendation systems intended to provide personal recommendations for various types of services and products, including:

- Web pages, e.g. (http://my.yahoo.com)
- Books, e.g. (http://www.amazon.com)
- TripAdvisor, e.g. (http://www.tripadvisor.com)
- Movies, e.g. (https://www.netflix.com), (http://www.moviefinder.tv)

Role of Recommender systems in academic D-library:

- To satisfy the users' requirements and their needs.
- To access relevant information very quickly.

The recommender system architecture usually comprises of (a) background data, which is the information the system has been generating recommendations earlier, (b) input data, the information that has to be entered in order to begin the process of recommendation, (c) an algorithm that combines the background data and input data to produce recommendations (Burke 2002). Classification of recommender systems was suggested based on how recommendations are made. There are three main types of recommender systems, namely, content-based filtering, collaborative filtering and hybrid filtering (Balabanovi and Shoham 1997). In addition to these three types, there are two other types that have been used to perform recommendations; they are Demographic based recommendation and Knowledge-based recommendation.

2.2.3.2 Recommender Systems Classes

2.2.3.2.1 Collaborative Filtering Approach

Sometimes called the social-based approach (Wan-Shiou, Jia-Ben et al. 2006) or user-to-user correlation approach (Li, Gu et al. 2009). In this approach, the system collects and analyse a large amount of information on users' preferences and determine recommendations to a target user based on their similarity to other users who are called nearest-neighbour (Herlocker, Konstan et al. 2004; Bobadilla, Ortega et al. 2012). The ''nearest-neighbour'' users are those that exhibit the strongest similarity to the target user and they act as ''recommendation partners'' for the target user (Bell and Koren 2007). Systems that are based on collaborative filtering approach depend only ontheitem and user identifiers and ignore user attributes (e.g., demographics) and item attributes (Schein, Popescul et al. 2002; Linden, Smith et al. 2003). Research on collaborative filtering can be grouped into two methods: memory-based and model-based (Christidis and Mentzas 2013). Memory-based methods (also called neighbourhood-based) make similarity comparison across the entire user's historical database to find out the most similar users to the active user and then, recommendations are generated based on the similar users' rating. Notable memory-based algorithms include the Pearson correlation algorithm (Popescul, Ungar et al. 2001), the Vector Space Similarity algorithm (Breese, Heckerman et al. 1998), and the Extended Generalized Vector-space algorithm (Soboroff and Nicholas 2000). Different from memory-based methods, model-based methods require constructing a descriptive model of users using the user-item ratings,and then recommendations are predicted using this descriptive model that can estimate the unknown ratings of a user. The common methods of this type include

“Regression Analysis”, “Association Rule”, and “Bayesian Network”. Memory-based and model-based algorithms have two kinds of approaches: item-based (Miller, Konstan et al. 2004), user-based (Herlocker, Konstan et al. 1999). The item-based approach recommends items based on its similarity to the ones the active user preferred in the past. The user-based approach analysesalarge amount of information on users’ preferences and determines recommendations to a target user based on other users that have similar interests (Im and Hars 2007; Xingyuan 2011). In university digital library scope, collaborative approach allows users to share experiences, in a way users can receive information that other users with similar profiles may consider useful (Tejeda-Lorente, Bernabé-Moreno et al. 2014).

The collaborative approach has the capability to recommend items that are not limited to similar items that other users have liked in the past (Li, Gu et al. 2009). In addition, it does not depend on the availability of textual descriptions (Billsus and Pazzani 1998). Therefore it has the capability of recommending complex items such as movies without requiring an "understanding" of the item itself.

Since the collaborative approach is based on the user activities they have the following limitations:

1. New items cannot be recommended to users until they have been rated by others. This problem is called the first-ratter problem or cold-start problem (Ahn 2008; Lika, Kolomvatsos et al. 2014).
2. Rating a very small portion of items by users leads to the lack of overlap of preferences between users and therefore makes it difficult to define neighbourhoods. This problem is called the Sparsity problem (Roh, Oh et al. 2003; Li, Gu et al. 2009).

3. As the number of users and items increases the computation time to calculate the similarity between users grows linearly resulting in poor scalability (Sarwar, Karypis et al. 2000).
4. The approach is biased towards the most popular items, i.e., items which have been rated by many users are more likely to be recommended than items that have few ratings (Adomavicius and Tuzhilin 2005).

2.2.3.2.2 Content-Based Filtering Approach

Also called Item-to-item correlation approach (Li, Gu et al. 2009). This approach recommends an item to a user based on item’s features and a profile of the user’s interests (Pazzani and Billsus 2007). The content-based approach requires textual descriptions of the items to be recommended (Billsus and Pazzani 1998), and it doesnot require other users' preference like the collaborative approach (Schein, Popescul et al. 2002). Content-based recommendation approach is used in a variety of domains: web pages, news articles, and items (Pazzani and Billsus 2007).

The content-based approach has the following advantages (Dong, Tokarchuk et al. 2009):

1. Does not suffer from the cold-start problem.
2. Does not suffer from the sparsity problem.

The content-based approach has the following limitation:

1. It can only recommend items matching the user’s past preferences. Thus, the recommended items will be the ones a user already knows (Shardanand and Maes 1995; Lops, de Gemmis et al. 2011).
2. It ignores the popularity of items (Dong, Tokarchuk et al. 2009).

3. In the domain of media including sound, picture, and video the approach faces the problem calculating the similarity among items (Demovic, Fritscher et al. 2013).
4. It is difficult for most content-based methods to find out the relationship between different names but describing the same item, i.e. many items have different names in real life. This problem is called the Synonymy problem (Shardanand and Maes 1995).

2.2.3.2.3 Demographic Filtering Approach

This approach provides recommendations based on the active user's demographic information (Pazzani 1999). Its advantage is that the user's history data is not needed, so a new user can obtain recommendation (Burke 2002; Yuanyuan, Chan et al. 2012). Since the Demographic approach is based on the demographic user's profile if the active user's demographic information is not available, it is not possible to recommend items to the active user (Adomavicius and Tuzhilin 2005).

2.2.3.2.4 Knowledge-Based Filtering Approach

This kind of approach recommends objects based on inferences about users' preferences and needs (Burke 2002). This approach sometimes provides explicit knowledge about how the recommended items meet the users' preferences (Tejeda-Lorente, Porcel et al. 2014).

2.2.3.2.5 Hybrid Filtering Approach

This approach combines multiple recommendations approaches together to produce its output. Using a hybrid approach helps to avoid certain limitations of recommendations approaches and give more effective recommendations in some cases (Adomavicius and Tuzhilin 2005; Porcel and Herrera-Viedma 2009). The most common hybridizing methodology is combining content-based approach and collaborative approach. The limitation of this approach is that it demands more information compared to the content-based approach or collaborative approach (Li, Gu et al. 2009).

It has been observed that the recommendation systems based on collaborative approach generally achieve more effective recommendations than systems based on content-based approach (Hwang, Hsiung et al. 2003; Hwang and Chuang 2004). A collaborative approach is preferred in environments where the number of users with accounts and activities on the site is high (Akbar, Shaffer et al. 2014), while the content-based approach is preferred in environments where interaction between users is low (Porcel, Moreno et al. 2009; Rajagopal and Kwan 2012).

2.2.4 Data Mining (DM)

Data mining is an information technology emerged as the development of database technology and artificial intelligence technology. Nowadays we are constantly bombarded with data in all scopes of our lives. Therefore, we have serious problems with accessing the relevant data. This problem caused by the rapid advances made in information and communication technologies (Edmunds and Morris 2000; Savolainen 2007). Information overload can be defined as “The inability to extract needed knowledge from an immense quantity of information” (Nelson 1994). The problem of information overload introduces noise in information access processes,and it affects making decisions. Information overload is a fundamental issue regarding the efficiency of using the data mining. Data mining is the process of extracting implicit and unknown information with the potential applicative value from the large amount of incomplete, noisy, random, and fuzzy data (Krishnamurthy and Balasubramani 2014). Data Mining has become an important area in information systems because it helps in analysing data from different perspectives and summarizing it into useful information. According to the different forms of the main data structure, data mining can be divided into three categories: data mining, text mining and web mining (Kantardzic 2002).

2.2.5 Web Mining (WM)

World Wide Web (called Web) is a popular and interactive medium to collect, disseminate and access an increasingly huge amount of information. This great amount of information introduces noise into information which affects decisions making (Tejeda-Lorente, Porcel et al. 2014). This problem stimulates the development of effective techniques that support analysing Web data. Web mining is the application of data mining technology to Web data handling Web documents, Web links, and Web log data. It is the process that extracts hidden and interesting knowledge from the World Wide Web (Singh and Singh 2010). It is a comprehensively integrated technique, involving Web technology, computer language, artificial intelligence, information retrieval, computer linguistics, informatics and statistics (Zhou and Le 2007). At present, Web mining technology covers various kinds of application tasks clustering, classification, association rules discovery, deviation checking and sequential pattern analysis (Mustafa and Kumaraswamy 2014).

Web involves three diverse types of data (Nassar and Al Saiyd 2013):

1. Heterogeneous Data content on the WWW
2. Hyperlinks exist among Web pages within site and across different sites.
3. The weblog data concerning the users who accessed the Web pages.

According to Web data types and analysis targets, Kosala and Blockee classified Web mining into three categories: Web content mining, Web structure mining and Web usage mining (Kosala and Blockeel 2000). The following paragraphs define the categories of Web mining:

- Web content mining is the process of discovering meaningful knowledge from Web pages' contents or descriptions. Web content is a very rich information resource consisting of many types of Web information, such as unstructured text, documents, audio, video, images and metadata (Gupta, Sharma et al. 2014).
- Web structure mining is used to discover potential web link structures and relations of Web pages. This process operates on Web pages' hyperlinks.
- Web Usage Mining: The focus of this study is on web usage mining. The term "Web Usage Mining" raised by Cooley in 1997. Web usage mining is the application of data mining techniques to discover and analyse the usage patterns of Web pages found among users visiting a Website (Mobasher, Dai et al. 2002; Malik and Rizvi 2011). The mined data often contain data logs of users' interactions with Web. The logs include information about the referring pages, user profiles, time a user spends at a site, user sessions or transactions, cookies, user queries, bookmark data, mouse scrolls,thesequence of pages visited, and any other data as the result of interactions (Jespersen, Thorhauge et al. 2002). The goal of Web usage is to capture, model, and analyse the behavioural patterns and profiles of Web users interacting with a Web site which helps to provide personalized items and services to them (Brigitte, Marie-Aude et al. 2008). Also, it can improve interactive dialogues by analysing users' click-stream sequences (Zanker, Fuchs et al. 2008). In addition, it improves the Web server performance (Pei, Han et al. 2000). While Web structure mining shows that page X has a link to page Y, Web usage mining shows who or how many people took that link, which site they came from and where they went when they left page Y (Gupta, Sharma et al. 2014). Web usage mining can provide personalized services for Web users by extracting and classifying the behaviours and interests of Web users. There are three main sources which can provide data for Web usage mining are as follows (Pani 2011; Eltahir and Dafa-Alla 2013):

1. Web Server logs: Log file record contains all information about the visitor's activity. Typical data includes IP address, page reference and request date/time. The common server log file types are access log, error log, agent log and referrer log (Suneetha and Krishnamoorthi 2009). The Web server log data present the most significant used source for web usage mining (Varnagar, Madhak et al. 2013).
2. Proxy server logs: At many places, internet services are routed through a dedicated machine known as a proxy server. The proxy server may serve as a data source to discover the usage pattern of users (V.Chitraa and Davamani 2010).
3. Client Logs: Client logs file record activities that happen within the client machine, for example, mouse wheel rotation, scrolling within a particular page, mouse clicks, and content selection (Choi and Geehyuk 2009).

The Web has had an impact on every aspect of society including commerce, science, government, and health (Varnagar, Madhak et al. 2013). Table 2.1 shows the facts of websites increase from June 2000 to October 2015 (Netcraft 2015). The numbers of services are increasing rapidly through the internet. Billions of users are using these services. Hence, users expect more intelligent systems to meet their needs. This has prompted the need for using techniques to discover knowledge on W.W.W to enhance decision making which adds value to business success and individuals, thereby giving rise to the term 'Web mining' (Wang and Ren 2009). Web mining has become the area of growing significance because it helps in discovery of information from the World Wide Web (Alam 2011).

Web Site Survey **Month & Year**	**Number of websites** **across all domains**
June 2000 till May 2005	The growth was observed from 7.542.571 to 29.407.337
June 2005 till May 2010	The growth was observed from 29.480.249 to 84.193.455
June 2010 till October 2015	The growth was observed from 86.832.845 to 171.009.994

Table 2.1: Increase in the websites from June 2000 to October 2015

Knowledge is an important resource for organizations to achieve competitive advantages. Knowledge discovery can be defined as the nontrivial extraction of implicit, previously unknown, and potentially useful information from massive data (Lomotey and Deters 2014). Nowadays, with rapid development in Web-based applications and artificial intelligence knowledge discovery in Web have been getting more attention (Gupta, Sharma et al. 2014). Web-based applications include massive amounts of data which present an interesting opportunity for knowledge discovery. To extract this valuable knowledge, there is a need for analytic methods for Web-based applications. Web mining technologies are the right solutions for knowledge discovery on the Web (Boddu, Anne et al. 2010). In fact, Web mining has been developed under the assumption that the Web's data could be used more profitably. Web mining is used to identify potential knowledge from mass Web data fortheenterprise that depends on Web for providing services (AGUILAR 2009). This knowledge can be used to aid business in making better decisions and raise the performance of Web services (Gupta, Sharma et al. 2014).

2.3 Utilization of Data Mining in Digital Libraries

With the increasing information resources in digital libraries, there ismore complex information to be handled and provided by libraries (Dianjun and Min 2011). Therefore, libraries made use of information technology to understand the needs of users and then provide the services accordingly (Dianjun and Min 2011). Since data mining is a very popular technique for analysing user's demands, there has been intense interest in how data mining can be used to support library management (Krishnamurthy and Balasubramani 2014). The authors (Li and Chen 2008) mention that, applying data mining techniques in the library system will provide effective support for library management. The data mining field specific to theanalysis of library data is called Bibliomining, the term was first brought by Nicholson in 2003 (Nicholson 2003). Bibliomining is the process of applying data mining techniques tolarge amounts of library associated data to extract patterns to increase the efficiency of services (Nicholson 2003; Azam, Sohrawardi et al. 2013). Nicholson defined a conceptual framework for Bibliomining(Nicholson 2006), which includes five basic elements:

1. Operation of the library.
2. Bibliographic records: Bibliographic records usually include the title of the item, its author, abstract, keywords.
3. Bibliometric data: Bibliometric data include citations and cross-references.
4. Library services: Library services usually include searching and circulation of resources.
5. Demographic structure of users, i.e., membership in interest groups, education.

Data stored in library systems include resource dataset, resource transaction, user profiles, access patterns, querying operations, retrieval operations, etc. Through applying data mining process to these data, a rich knowledge base withahigher value will be available (Donnellan and Pahl 2002; Romero and Ventura 2007). This knowledge providesdeeper understanding of users which helps to take precise decisions for providing good services to them (Krishnamurthy and Balasubramani 2014).

Data mining techniques can potentially be deployed in a web-based digital library system to analyse its data. Its techniques have great application space and value in the digital library systems (Sitanggang, Husin et al. 2010). Zhang noted that the use of data mining technology providesa powerful safeguard for theWeb-based digital library to make effective decisions (Zhang 2011). Krishnamurthy and Balasubramani added that Web data mining techniques help the librarians for providing better services and effective utilization of resources (Krishnamurthy and Balasubramani 2014). In (Romero and Ventura 2007; Zhang 2011) the authors mentioned that web mining became a promising technology in digital library management. The following systems present a number of data mining applications in D-libraries:

- **Personalization Recommender Systems**

  Personalization recommender systems aim to find resources that match library users' interest (Krishnamurthy and Balasubramani 2014).

- **Scholar Searching Systems**

  Scholar searching systems recommend important researchers of a research topic (Juan, Kejun et al. 2009).

- **Library Budget Allocation Systems**

  Library budget allocation system ensures the availability of the needed resources (i.e.,Some resources in the library may not be needed by users while others resources may be needed by the users but not available). Resource purchase plans are mostly determined by library staff just depending on their experience (Ma and Xiao 2010). Data mining can be used asguidance of library buyer. Data mining techniques analyze circulation statistics to reflect the future demands (Wu 2003; Dinkins 2011).

- **Disseminate Indigenous Knowledge Systems**

  Systems used to keep users informed of new resources on specified topics. Whenever new information is entered into the system, the interest profiles are used to determine which information should be delivered to which users (Bertino, Ferrari et al. 2001).

- **Optimizing Search**

  The optimized searching application provides users with what they want precisely (Bhide, Yoo Jae et al. 2007).

- **Readers' Reading Tendency Systems**

  Data mining technology has the ability to discover changeable interest of readers automatically (Li and Chen 2008). Readers' reading tendency systems aim at analyzing readers' reading habits to understand the utilization of each subject category and predict future user needs (Fan, Ya-Han et al. 2011; Uppal and Chindwani 2013).

- **Subject Headings Systems**

  Subject headings system is the system that discovers associations between the book categories. This could be used to predict books for users. Also, it could help to make better shelving decisions.

From the previously mentioned applications, data mining is an important need for digital libraries to bring out the hidden knowledge that allows digital libraries to provide effective management functions resulting in high user satisfaction and high efficiency of resource utilization. There are a number of studies reported in the literature that used data mining techniques in academic library management system for a variety of applications. Table 2.2 shows some studies that utilize data mining technology in the D - library domain.

Application domain	Study	Data mining category	Data mining technique
Personalized recommendation	- Research on intelligent recommended algorithms of personalized digital library (Huaxin and Qian 2013)	Data mining	Association rule algorithm
	- Web mining application in university library personalized search engine (Zhao 2011)	Web mining	Clustering algorithm
	- Sequential pattern mining on library transaction data (Sitanggang, Husin et al. 2010)	Data mining	Sequential pattern algorithm
	- Research on personalized recommendation based on web usage mining using collaborative filtering technique (Wang and Ren 2009)	Web mining	Clustering algorithm
	- Personalized Links Recommendation Based on Data Mining in Adaptive Educational Hypermedia Systems (Romero, Ventura et al. 2007).	Web mining	Clustering & Sequential pattern algorithm
	- A prototype WWW literature recommendation system for digital libraries (Hwang, Hsiung et al. 2003).	Web mining	Association rule algorithm
	- Using adaptive resonance theory and data-mining techniques for resources recommendation based on the e-library environment (Tsai and Chen 2008).	Data mining	Artificial neural network
Scholar Searching	- Using Web-Mining for Academic Measurement and Scholar Recommendation in Expert Finding System (Chi-Jen, Jen-Ming et al. 2011).	Web mining	Association rule algorithm
Navigation Recommendations	- Data mining analysis of digital library database usage patterns as a tool facilitating efficient user navigation (GIBSON 2001).	Web mining	Association rule algorithm & Clustering algorithm

Library budget allocation	• Budget allocation model for the academic library acquisition using data mining technique (Hossain and Rahman 2014).	Data mining	Decision tree-ID3 algorithm
	• An Association Rule Mining Approach for Libraries to Analyse User Interest (Krishnamurthy and Balasubramani 2014).	Data mining	Association rule algorithm
	• K-means clustering algorithm application in university libraries (Runhua, Yi et al. 2011).	Data mining	Clustering algorithm
	• Decision support for the academic library acquisition budget allocation via circulation database mining (Kao, Chang et al. 2003).	Data mining	Decision treealgorithm
Collaborative learning	• A graph-based data mining method for collaborative learning space in learning commons (Okamoto, Asanuma et al. 2014).	Data mining	Association rule algorithm & Clustering algorithm
Readers' reading tendency	• Exploring New Trends of University Libraries by SPSS Cluster Analysis Method (Chen 2011).	Data mining	Clustering algorithm
	• Application of data mining in the analysis of needs of university library users (Tingting and Lili 2011).	Data mining	Clustering algorithm
	• Enhancing library resources usage efficiency by data mining (Tung-Shou, Ming-Horng et al. 2004).	Data mining	On-line Analytical Processing mining
Subject headings	• Borrowing Data Mining Based on Association Rules (Xiaojian and Yuchun 2012).	Data mining	Association rule algorithm
	• Bibliomining on North South University library data (Azam, Sohrawardi et al. 2013).	Data mining	Association rule algorithm

Table 2.2: Utilization of data mining in Digital libraries

2.4 Methods used to Alleviate the Sparsity Problem

Many different methods and techniques have been proposed used with collaborative filtering to deal with the sparsity problem, including dimensionality reduction, hybrid approach with content-based, hybrid approach with demographic information, utilizing implicit feedback, and shared collaborative filtering approach.

Dimensionality reduction techniques. Dimensionality reduction techniques address the sparsity problem by removing information that is not informative for the task from the rating matrix so as to condense the matrix and therefore can be used much more efficiently than can the original sparse matrix.

Hybrid approach with content-based. Researchers have attempted to combine content-based approach with collaborative filtering to alleviate the sparsity problem. In addition to user ratings, content-based approach considers similarities between items, thus to make accurate predictions.

Hybrid approach with demographic information. In addition to user ratings, demographic information is used when calculating user similarity. This creates an additional entity connected to the users than simply just user-item ratings. In such hybrid approach, users are considered similar if they rate the same items,orthey belong to the same demographic group.

Utilizing implicit feedback. The idea of utilizing implicit feedback is to decrease the dependency on the user's explicit rating and increase the volume of feedback to the rating prediction. Through the observation made of users' activities (such as their login times and history of viewed items), implicit feedback became an important source for exploiting knowledge about user preferences.

Shared collaborative filtering approach. Lately, researchers presented the shared collaborative filtering approach (also known asacross-domain recommendation) for alleviating the data sparsity problem. The aim of this approach is to transfer rating data from other domains, referred to as the auxiliary domain, to sparse rating dataset in a target domain to decrease the impact of sparse rating on the prediction results in the target domain.

The mentioned methods and techniques in this section have presented their success in alleviating the effect of data sparsity. However, such methods have some limitations. Table 2.3 shows these limitations:

Method	Limitations
Shared collaborative filtering approach	• The rating in both target and auxiliary rating sources may rarely have the same rating pattern. • Not easy to find an auxiliary data source contains users and items similar to the target domain users and items. • View domains can reflect similar aspects.
Hybrid approach with demographic information	• Users' demographic information should be updated periodically otherwise the system will give static suggestions. • Compromise the privacy of users.
Hybrid approach with content-based	• In many cases, the items' information is not enough formodelling the items. • Requires metrics to compute similarities among the items. Such similarity metrics are expensive to acquire.
Dimensionality reduction	• Due to the process of finding the low-dimensional subspace, some useful information might be removed during this process.
Implicit feedback	• Most observable actions provide only positive feedback. • The quality of some observable actions might be lower (Ex. Long read time do not indicate high interest, it's more likely that some users need more time for viewing an object than others or a user can forget to close the browser window) • Lack of granularity of implicit rating, which makes it hard to estimate the degree to which a user prefers an item. • Most systems based on implicit feedback limited to the binary rating. Binary implicit feedback has only two values, 1 which indicates that the user likes the item. The other value is 0. Zero can indicate that the user likes the item or the user didn't view the item. However, when computing similarities, it is not known if the zeroes are blanks where the user has not view the item (missing values), or it is a negative preference.

Table 2.3: Limitations of methods used for alleviating the sparsity problem

2.5 Related Works

In this section, the use of collaborative filtering approach in academic D-library will be reviewed and compared to the study work.

Despite all the limitations of the collaborative filtering approach, this approach is by far the most used approach in library recommender systems (Hwang and Chuang 2004; Ahn 2008; Bobadilla, Ortega et al. 2013). This approach allows users to share their experiences with the community so that items found useful by others with similar profiles can be recommended (Leung, Chan et al. 2008; Porcel, Castillo et al. 2010). Based upon the two major methods of collaborative filtering that is memory based and model-based many studies had been carried out to develop D-library recommendation system (Avancini and Straccia 2005; Chen and Chen 2007; Zhu and Wang 2007; Li and Chen 2008; Mönnich and Spiering 2008; Porcel and Herrera-Viedma 2010; Sitanggang, Husin et al. 2010; Lina and Zhiyong 2013; WeiJi, Liu1 et al. 2016).

Porcel and Herrera-Viedma (Porcel and Herrera-Viedma 2010) developed a fuzzy linguistic recommender system for research resources. The system first builds the user's profile by requesting users to provide their preferences over a limited number of research resources. The user profile is composed of: a) User's preferences on topics, b) User's preferences on collaboration possibilities with other users. The system recommends resources to users by calculating the linguistic similarity measure between users, and then the user's profile is completed with user's preferences on the collaboration possibilities with compatible users. However, despite its success to avoid the information overload, there is a large amount of resources daily added to the library, this decrease the system performance. In addition, the system can become computationally expensive, in terms of time as the number of users increases.

Recommendation systems based on association mining are very popular in D-library systems. The studies (Li and Chen 2008) and (Zhu and Wang 2007) are based on association rule mining. It analyses the historical user data and then uses the mined rules for recommending resources to the active user. Li and Chen (Li and Chen 2008) proposed a model based on Apriori algorithm to extract association rules from book reader's lend records. The Apriori algorithm is the classical algorithm of association rules (Agrawal and Srikant 1994). The system firstly starts a data preparation process, it seeks for the 50 most popular books based on books' lending statistics, and then search for the most related books for the 50 selected books from the lending records. After the active user borrows a book, then a list of books which are frequently borrowed together with the borrowed book is recommended. The Apriori algorithm used by (Li and Chen 2008) have some shortcomings such as that a large number of candidate items are generated, and much more disk input/output operations are needed. In addition, it is in very slow for large databases because the algorithm scans the database multiple scans to determine which candidates are frequent. Thus it will take more memory and time (Mahajan, Pawar et al. 2014; More and Somaiya 2014; Arivazhagan and Pragaladan 2015). In (Zhu and Wang 2007) the authors have proposed a recommendation model based on an improved Apriori algorithm that discovers knowledge from library circulation records to recommend books. The proposed recommendation model follows two methods to improve the efficiency of Apriori algorithm. The first method used to reduce the candidate item set by applying selective factors and by deleting the items that do not satisfy minimum support. The second method is to ignore the transaction records that are useless for frequent items generated. By applying these two methods, the database size will be decreased so that the time needed to scan transactions can be reduced.

In addition to using association rule mining in recommendation systems, sequential pattern mining technique was applied in (Sitanggang, Husin et al. 2010) to recommend books to students. The study applied a well-known sequential pattern mining technique namely, AprioriAll proposed by Agrawal and Srikant (Agrawal and Srikant 1995). The study applies the AprioriAll algorithm on the historical circulation records to find out the most frequent-borrowed book sequences of readers, and from this information the system provides recommendations to students after they borrow a book. The algorithm consists of five phases: sort phase, litmset (large item set) phase, transformation phase, sequence phase, and maximal phase. In the sort phase, the dataset is sorted by using two attributes, the first attribute is the user ID as the major key and the second attribute is the transaction time as the minor key. This phase results a database of a sorted user sequences. In the large itemset phase, all frequent item-sets with minimum support in the database are identified. In the transformation phase, each transaction contained in a user sequence is replaced by all large item-sets contained in that transaction based on two conditions: transactions that do not contain any large item set are not retained in the transformed sequence and sequences do not contain any large item set are dropped from the transformed databases. In the sequence phase, Count-all method based on the Apriori algorithm is used to find all large frequent sequences. In the maximal phase, the maximal sequences among the set of large sequences are identified. Because the complexity of AprioriAll algorithm is so high and it has low-speed analysis (Mabroukeh and Ezeife 2010; Nguyen and Nguyen 2012) the system suffers from increased delays, especially when applied in a huge database that contains very long frequent sequences. It has the advantage of the ability to discover interesting patterns over time which lead to more accurate recommendations (Bonnin and Jannach 2013).

The studies (Lina and Zhiyong 2013) and (Chen and Chen 2007) used clustering mining and association rule mining to analyse library data and then used the mined rules for the recommendation process. The purpose of using the clustering technique is to reduce the amount of data involved in the association rule mining to remove the books which are borrowed by low frequency, subsequently save the scanning time and improve the quality of mining. In (Lina and Zhiyong 2013) the system process begins by using K-means clustering technique to cluster the behaviour of borrowing books. The K-means algorithm divides the reader's borrowing behaviour into three classifies high frequency, medium frequency, and low frequency. Then the Apriori algorithm is applied to the books which are borrowed by high and medium frequency readers. When an active user requests a book, the system recommends books which match the association rules. In (Chen and Chen 2007) the system composes two phases. In the first phase, Ant Colony Clustering Algorithm (Dorigo and Gambardella 1997) is used to group lend records according to demographic information. In the second phase, the Apriori Algorithm is used to analyse the relation between borrowed books in each cluster and find their association. Then the association rules are used as the basis of the book recommendation process. The two studies (Lina and Zhiyong 2013) and (Chen and Chen 2007) benefit from clustering, clustering will reduce the search space, thus computing recommendations will be faster than scanning the entire database, and predictions for recommendations are computed independently within each cluster which improves the quality of recommendations.

In (WeiJi, Liu1 et al. 2016) they propose a book recommender system of university library based on the k-nearest neighbour algorithm and Association Rules Mining. The k-nearest neighbour algorithm uses cosine similarity to find similar students based on demographic information. After getting the readers nearest neighbours the Apriori algorithm is applied to extract association rules from the nearest neighbours borrowed books, and then the association rules are used to produce the recommendation list.

Karlsruhe University in Germany has developed a recommender system called BibTip funded by the German Research Foundation (Mönnich and Spiering 2008). The BibTip recommender system is based on the observation of user patterns during the catalogue search. The approach taken by this recommender system will be to define a co-occurrence user-item matrix based on the selection of titles within defined sessions. The co-occurrence matrix gives the number of times the items co-occur in an observed data set. The system consists of three software agents:

The Observation Agent, the Aggregation Agent, and the Recommendation Agent. The Observation Agent observes the selection of titles during sessions in the online catalogue; these titles are transferred to the Aggregation Agent which collates co-occurrences resources. These co-occurrences resources are represented in a co-occurrence matrix, recording which resources appear together in user histories, and then, given a matrix based on the running totals of resources pairs each pair of across all users. The Recommendation Agent provides a list of recommendations to users based on evaluating the co-occurrence matrix.

Avancini and Straccia propose the system CYCLADES (Avancini and Straccia 2005). The system provides the folder-based environment to support collaboration between users with similar interests by way of folder sharing. Each user has their own folder; this folder contains metadata explaining what the folder contains which represent the set of user's interest topics.

The details of recommender systems reviewed above are summarized in Table 2.4, including its recommendation field, data type, information required for recommendation process, the applied techniques used fortherecommendation, are presented in the table. Despite that the use of recommender systems reviewed above avoided the information overload problem, there are different aspects that may affect their accuracy and performance. The advantages and disadvantages of these systems are presented in Table 2.5.

Study	Method type	Recommendation field	Data type	Information Required	Technique
Research of Intelligent recommendation system based on the user and association rules mining for books (WeiJi, Liu1 et al. 2016)	Memory-based	Books	Structure data	Users' lend records, users' demographic information	k-nearest neighbour (cosine similarity), Association rule mining (Apriori algorithm)
The Application of Book Intelligent Recommendation Based on the Association Rule Mining of Clementine (Lina and Zhiyong 2013)	Model-based	Books	Structure data	User's lend records	Cluster mining (K-means algorithm), Association rule mining (Apriori algorithm)
Sequential pattern mining on library transaction data (Sitanggang, Husin et al. 2010)	Model-based	Books	Structure data	User's lend records	Sequential pattern mining (AprioriAll algorithm)
Dealing with incomplete information in a fuzzy linguistic recommender system to disseminate information in university digital libraries (Porcel and Herrera-Viedma 2010)	Memory-based	Journal articles and books	Structure data	User preferences	Multi-granular Fuzzy Linguistic Modelling
The application of Association rule in Library system (Li and Chen 2008)	Memory-based	Books	Structure data	User's lend records	Sequential pattern mining (AprioriAll algorithm)

Study	**Method type**	**Recommendation field**	**Data type**	**Information Required**	**Technique**
Adding Value to the Library Catalog by Implementing a Recommendation System (Mönnich and Spiering 2008)	Memory-based	Resources	Log data	Users' usage data	Co-occurrence matrix
Using data mining technology to provide a recommendation service in the digital library (Chen and Chen 2007)	Model-based	Books	Structure data	Users' lend records, users' demographic information	Ant Colony Clustering Algorithm, Association rule mining (Apriori algorithm)
Book Recommendation Service by Improved Association Rule Mining Algorithm (Zhu and Wang 2007)	Model-based	Books	Structured data	User's lend records, User's demographic information	Association rule mining algorithm
User recommendation for collaborative and personalised digital archives (Avancini and Straccia 2005)	Memory-based	Resources	Log data	Explicit rate, historical usage of resources, user activity	k-nearest neighbour (Pearson correlation coefficient)

Table 2.4: Survey of Collaborative approach in academic D-library

Study	Advantages	Disadvantages
Research of Intelligent recommendation system based on the user and association rules mining for books (WeiJi, Liu1 et al. 2016)	• Alleviate the cold-start problem. • Can provide recommendations without the user provide any data.	• The users' demographic information should be updated periodically otherwise the system will give static suggestions.
The Application of Book Intelligent Recommendation Based on the Association Rule Mining of Clementine (Lina and Zhiyong 2013)	• Improve both the execution time and accuracy by: 1. Removing the books which are borrowed bythelow frequency. 2. Clustering the database which will reduce the search space.	• Generate recommendations based on the most borrowed books.
Sequential pattern mining on library transaction data (Sitanggang, Husin et al. 2010)	• Provide implicit information (show student's behaviour in borrowing books). • Considered the time in the mining process produce more accurate recommendations.	• Has a higher computational complexity.
Dealing with incomplete information in a fuzzy linguistic recommender system to disseminate information in university digital libraries (Porcel and Herrera-Viedma 2010)	• Deal with the incomplete user's preference thus reduces the user effort to characterize their user profiles. • Alleviate the Cold start problem.	• Users provide their preferences explicitly • Computationally expensive in terms of time as the number of users increases.
The application of Association rule in Library system (Li and Chen 2008)	• Reduce the execution time by grouping books based on the most lending books	• Restrict the user preferences over 50 books. • Sparser rules impact the quality of the recommendations.
Adding Value to the Library Catalog by Implementing a Recommendation System (Mönnich and Spiering 2008)	• Provide dynamic recommendations based on the active user's activity.	• The computational complexity is high.

Study	Advantages	Disadvantages
Using data mining technology to provide a recommendation service in the digital library	• Grouping the borrowing books according to users' demographic information reduce the execution time.	• The users' demographic information has to be updated periodically otherwise the system will give static suggestions.
Book Recommendation Service by Improved Association Rule Mining Algorithm (Zhu and Wang 2007)	• The time spent on searching for frequent itemsets is reduced • The I/O load is reduced	• Number of rules generated are based on the number of itemsets in a frequent item sets. • Sparser rules impact the quality of the recommendations.
User recommendation for collaborative and personalised digital archives (Avancini and Straccia 2005)	• Generating cross-library recommendations. In that way, libraries with low usage can provide recommendations to their users. Also, a sufficient data for generating recommendations can be collected quickly. • Using previous use of resources as implicit rating	• It provides a collaborative digital library environment more than it is a recommendation system.

Table 2.5: Advantages and Disadvantages of previous related work

The goal of all the studies mentioned above is to provide accurate recommendations. The proposed framework differs from the above related studies:

1. Their recommendation approach (Li and Chen 2008; Lina and Zhiyong 2013) suggest resources only related to a limited number of resources based on the most borrowed books; whereas the proposed framework covers all resources outside the scope of the most borrowed resources.
2. Thestudy(WeiJi, Liu1 et al. 2016) classify users according to their demographic information. Recommender systems based on demographic information mustupdate the user demographic information regularly to keep up with the ever-changing user demographic information otherwise the system will give static suggestions according to what is contained in the user demographic information. In contrast, the proposed framework recommendations are dynamically adjusted to the changing usage of resources by the user. If users at one point begin to use resources in a different context than had been used before, this will be reflected in the lists provided by the system.
3. In (Avancini and Straccia 2005) the recommendation engine request users to provide explicit feedback through ratings of relevant resources which represent a tiring process for users. As a result, users rate only a small number of resources, so the accuracy of recommendation will be decreased due to sparse data. The proposed framework differs in that it does not require the user to state their preferences explicitly, instead, their preferences are inferred implicitly from the user's interaction with the system. As a result, plenty of data can be available for the recommendation engine.
4. These studies (Mönnich and Spiering 2008), (Li and Chen 2008), (Zhu and Wang 2007), (Chen and Chen 2007) use Apriori-based algorithms. Apriori-based algorithms have a large search space,and it is computationally expensive, that is it requires multiple scans of the database to determine which candidates are frequent (Mahajan, Pawar et al.

2014), while the proposed framework uses clustering algorithms which have a small search space.

5. These studies (Li and Chen 2008; Lina and Zhiyong 2013) introduced mining association rules for discovering interesting relations between items in a large transaction database. The association models in these systems used lend transactional data. Transactional data, by its nature, are extremely sparse. (Cadez, Smyth et al. 2001; Apte, Liu et al. 2002) mention that applying association rule mining using a parse transaction database to predict user behaviour may not be appropriate. In association rule mining every rule is composed by two different sets of items, X and Y where X is called antecedent (or left-hand side), and Y is called consequent (or right-hand side). Rule $X \rightarrow Y$ indicates that whenever an itemset X occur in a session, then the itemset Y also will occur in the same session. Then items in the antecedent are used to generate recommendations to the active user by matching the active user preferences against the items in the antecedent. As the similarity algorithms in systems based association rule mining tries to find rules that are similar to the active user profile when generating association rule mining from a sparse dataset, usually there will be a large number of active users' preferences don't match with the antecedent of the rules, due to parser rules. Using parser rules for identifying neighbours is misleading. However, in contrast to the proposedframework, their function is limited to alleviate the sparsity problem.
6. The system (Sitanggang, Husin et al. 2010) utilizes sequential patterns miningbecause sequential pattern mining isa sequential version of association rules. Thus their system suffers from the same problem of association rule mining which is the problem parser rules.
7. These systems (Li and Chen 2008; Mönnich and Spiering 2008; Porcel and Herrera-Viedma 2010) are memory-based collaborative filtering systems. In the presence of data,

sparsity sets common items between users are few. Therefore similarity values are unreliable,and their accuracy is very poor(Pinto, Tanscheit et al. 2012; Joshi1 and Paswan 2013). In contrast to this, the proposed framework is model-based collaborative filtering.

8. These systems (Avancini and Straccia 2005; WeiJi, Liu1 et al. 2016) based on k-Nearest Neighbours algorithms. The accuracy of nearest neighbours algorithms is very poor for sparse data (Demiriz 2004). In contrast to the proposed framework, their function is limited to alleviate the sparsity problem.

The above examples mentioned in the related work section are sufficient in revealing some major considerations and issues that impact the effectiveness of recommendation systems for D-library, as follows:

- Providing more feedback data add value to the recommendation results.
- Building model-based collaborative filtering must be considered as a key factor in alleviating sparsity problem.
- Grouping users into different categories based on preferences allow representing the information needs and alleviate the scalability problem.

2.6 Summary

This chapter has reviewed background knowledge and work related to the proposed framework. A definition of academic digital library and benefits that academic D-library can provide in education is presented. A detailed description of recommender systems categories is presented. An overview of the area of data mining, specifically on the topics of the Web usage mining is presented followed by the application of Data Mining in the field of digital libraries. Next, existing proposed solutions used to mitigate the sparsity problem, and the drawbacks of these proposed solutions are presented. Finally, previous work on collaborative recommender systems in the field of D-library has been presented and discussed.

CHAPTER 3

A Framework for Optimizing Academic D-Library Recommender Systems

3.1 Introduction

In the last few years, Web-based academic D-libraries are faced with large amounts of digital resources. This is the fact that more and more users are moving to the Web-based D-library (Okojie 2010), which forced Web-based academic libraries to start thinking about maintaining an effective decision service for identifying appropriate resources to users. Recommendation service is used in the design of academic digital libraries as a means to help users find appropriate resources within the large amounts of digital resources (Aijuan and Baoying 2008; Rajagopal and Kwan 2012).

The purpose of this study is to present an application of Web clustering mining technology to optimize recommender systems for online academic digital libraries. To achieve this purpose, this chapter provides development of a collaborative filtering framework that uses Web usage mining technology. The proposed framework incorporates the discovered usage knowledge from implicit feedback data with clustering technique. It has been recognized that web usage mining gave better recommendation quality in the collaborative filtering procedures

(Samizadeh and Ghelichkhani 2010; Suguna and Sharmila 2013 ; Lopes and Roy 2014). Recently, a variety of Web recommendation systems have built through web usage mining and collaborative filtering approach have been proposed and has achieved great successes (See, for example, On-line book retailer Amazon www.amazon.com, DVD rental provider Netflix www.netflix.com). Along with this line, the proposed framework can be seen as a Web recommender system that combines Web usage mining with the collaborative filtering approach.

The rest of this chapter is organized as follows. Section 3.2 discusses in details how the use of implicit feedback data and Web usage clustering mining technique alleviate the sparsity problem and can provide an effective collaborative filtering recommender system for D-libraries. Section 3.3 describes the proposed Web usage mining framework. The summary of this chapter is given in section 3.4.

3.2 Enhance Academic D-Library Recommender Systems through Web Usage Data

Authors in (Dixit, Gadge et al. 2010; Jalali, Mustapha et al. 2010; Adeniyi, Wei et al. 2014) mention that Web usage mining has great potential for the Web-based personalized recommendation. The use of Web usage mining techniques has been proven to be useful in many applications of recommender systems, e.g. e-commerce recommender systems (Peska and Vojtas 2013; Si, Sarma et al. 2014; Esmailian and Jalili 2015; Wu, Tan et al. 2015), social recommender systems (Nov and Arazy 2015), television and online video recommender systems (Neumann and Sayyadi 2015), tourism recommender systems (Bidart, Pereira et al. 2014; Yung 2015), E-learning recommender systems (McGrath 2008; Zorrilla, García et al. 2010). Many studies have focused on applying Web usage mining techniques in the domain of library recommendation systems, e.g. managing digital content (Suresh 2007), information overload (Porcel, Castillo et al. 2010), studying teachers' use of digital libraries (Xu and Recker 2011), increase the performance of library system (Ke, Kwakkelaar et al. 2002), and alleviate information overload (Xu, Zhang et al. 2005; AlMurtadha, Sulaiman et al. 2011). It has been recognized that web usage mining gave better recommendation quality in the Collaborative filtering procedures (Samizadeh and Ghelichkhani 2010; Suguna and Sharmila 2013 ; Lopes and Roy 2014). Web usage mining can play a prominent role to extract patterns and infer rules in Web-based academic D-library, which can help recommending relevant resources to users through the knowledge gained. The primary goal of using Web usage mining is to leverage implicit feedback data to build user model and also to use this model for resources recommendation. Models based on Web usage mining can gain better understanding of the Websites users (Mobasher 2007).

3.2.1 Inference Preferences from Web Usage Data

In a collaborative filtering recommender system, feedback is required to identify users' preferences. Implicit feedback takes advantage of user behaviour to generate relevance feedback to enrich the student profile. Implicit feedback has been used in different areas of collaborative filtering recommender systems for example, job recommender systems (Bradley, Rafter et al. 2000; Lee and Brusilovsky 2009), e-commerce recommender systems (Kim, Yum et al. 2005), restaurant recommender systems (Kuo, Wang et al. 2015), music recommender systems (Yang, Chen et al. 2012), news recommender systems (Muralidhar, Rangwala et al. 2015), movie recommender systems (Dez, Chavarriaga et al. 2010; Liu, He et al. 2013).

Pan and Scholz mentioned that implicit feedback seeks to avoid the problems of explicit feedback (Pan and Scholz 2009). Implicit feedback method provides several advantages for academic D-library compared to explicit feedback method:

a) Implicit user feedback is more reliable than explicit user feedback (Schafer, Frankowski et al. 2007). It conveys information regarding the user's preferences

(Radlinski and Joachims 2005; Peska and Vojtas 2015). Dingming et al. mention user behaviour identification is an important issue tounderstanding the users' preferences (Dingming, Dongyan et al. 2008). In D-library, every click students make on D-library website they leave behind pieces of information describing their preferences. Thus, it is suitable to use implicit feedback method to reflect student's interest.

b) Implicit feedback data can be obtained by observing the users behaviour without any additional burden on the users (Gauch, Speretta et al. 2007; Lops, de Gemmis et al. 2011; Volkovs and Yu 2015). Requesting feedback from users represent a major drawback of explicit feedback. Using implicit feedback is preferably for students because explicit feedback is expensive in terms of time and effort.

c) Achieve much greater coverage over resources (Jawaheer, Weller et al. 2014).

d) Implicit user feedback provides much more massive quantities of data rather than the sparseness encountered by explicit user feedback (Vellino 2010; Volkovs and Yu 2015). Usage data allows unlimited of implicit rate to be collected without requiring additional effort from users (Isinkaye, Folajimi et al. 2015). In D-library many different actions can be considered as implicit feedback. As a result,amassive amount of feedback can be gathered, subsequently, to alleviate data sparsity problem.

To suggest resources to students, it is essential to understand the students' behaviour and serve them accordingly (Krishnamurthy and Balasubramani 2014; Tejeda-Lorente, Bernabé-Moreno et al. 2014). Observation of student's interacting behaviour with the academic D-library Website can potentially generatea massive amount of implicit feedback data that can be used to identify students' preference. Common user behaviours that can been used by recommender systems as implicit feedback include purchasing, listening, bookmarking, saving, clicking links, spending time, scrolling a page, saving, forwarding, referencing, printing etc. For a review of sample of implicit feedback in D-library, see (Appendix A). In this study, five different student actions have been identified, including: click stream data (Smyth, Freyne et al. 2004; Xu 2008), the time spent on a page (Kelly and Belkin 2004; White, Jose et al. 2006), printing (Oard and Kim 1998), visiting order (Herlocker, Konstan et al. 2004), scrolling down on a Web page (Claypool, Le et al. 2001), repeated visits to particular type of item (Claypool, Le et al. 2001), and bookmarking (Oard and Kim 1998). These actions perform a strong correlation between them and the relevance of resources. The user's actions used for recommendation must consider different levels of evidence (Kelly and Teevan 2003). We assume that the actions identified in the proposed framework provide different levels of evidence of interest. The actions and their level of evidence are explained as follows:

Printing Action: The action print allows the users to print a resource. Cost plays an important role in determining the level of interest (Oard and Kim 2001). Print action costs the user more expensive resources(ink, paper ….) than using other actions, thus, printing a resource identify strong evidence for relevant resources than using "bookmark action" or "download action" or "read action". Printing behaviour can identify relevant documents (Oard and Kim 1998; Jinmook Kim, Oard et al. 2000).

Bookmark Action: The action bookmark allows the users to have the possibility to make further use of a resource. If a user found a resource is really interesting, they prefer to return to the resource another time. There is a strong correlation between bookmarking asasource of implicit feedback and the relevancy of documents (Seo and Zhang 2000). The authors (Fox, Karnawat et al. 2005) found that printing and bookmarking actions reflect high evidence for interest.

Download Action: The download action allows users to save the resource on their devices. Users need to be sure that the resource is relevant before downloading the resource, thus download action indicates evidenceof interest for resources.

Read Action: The read action also referred as view action allows users to view the details of a resource. Reading action is one of the most common sources of implicit feedback used by recommender systems. The time factor plays a major role in determining whether the resource is relevant or not when performing the read action. The action read is assumed to be interesting if the user spends time after a certain amount of time known as “read time threshold” (Seo and Zhang 2000). After the user spends the “read time threshold” we can determine that they found it relevant other it isn’t relevant. It is difficult to use the read action to determine whether the resource is relevant or not because it is difficult task to determine the “read time threshold” for the following reasons: it's more likely that some users need more time for viewing an object than others or a user can forget to close the browser window or may be the user is doing other activities while the resource is displayed, the lead to record long time for the read action, thus it ishard to determine the proper time for “read time threshold”. For this reason, it will take the lowest positive implicit weight proposed in this study. The study (Seo and Zhang 2000) evaluated the actions bookmarking and reading time as implicit measures of interest; they found that reading time was found less evidence of interest than bookmarking.

Abstract action: The “viewing abstract” action is primarily used to give an overview of the resource contents to users in order to check whether it is useful or not. This action is only useful for the recommendation process when used in combination with the other actions.

The above-mentioned actions can be used for two purposes. Printing, bookmarking, downloading, and viewing actions are treated as positive feedback which indicates that the student is interested in a resource. In contrast, “viewing abstract” action, it is rather difficult to tell, whether it is positive feedback or not. Thus, “viewing a resource abstract” is considered as a “check” action,i.e. the student is not sure that the resource is useful for them or not. If a student ignores a resource after viewing its abstract, it is more likely that there is something undesirable about the resource and the student is not satisfied with the resource, in this case, viewing the abstract of a resource is treated as anegative feedback.

To apply collaborative algorithms, the implicit feedback must turn into numeric ratings (Lichtenstein and Slovic 2006; Ekstrand, Riedl et al. 2011). For this purpose, a weighting schema is introduced to translate implicit feedback to numerical values. The weighting schema seeks to produce numerical ratings similar to those that a student would have explicitly assigned. Previous studies (Seo and Zhang 2000; Oard and Kim 2001; Fox, Karnawat et al. 2005) and a questionnaire participated by 35 students (from Bisha university - Saudi Arabia) are taken into consideration when defining the weighting schema (Appendix B). The action “print” provides the highest evidence of interest. In particular, if a user prints a resource then it is reasonable to conclude that they are convinced about the resource and can be assumed that, that resource is useful for the user. The action “download” comes second; the action ”bookmarks” comes third, if the resource was only viewed on the screen, we could have much less confidence. Therefore, the “read” provides the least evidence of interest. Table 3.1 presents the actions and appropriate weightings for each action representing the importance of it, where a higher value means higher evidence of interest. The numerical value ratings are between 1 and 5. The printing action received a weight of 5 (highest interest means student likes it best), bookmarking action received a weight of 4, downloading action received a weight of 3, reading the resource action received a weight of 2, viewing the abstract without performing one of the other actions received a weight of 1 (lowest interest means student doesn’t like the resource).

Action	Weight
Print	5
Download resource	4
Bookmark	3
View resource details	2
Ignore a resource after viewing its abstract	1

Table 3.1: Actions weights table

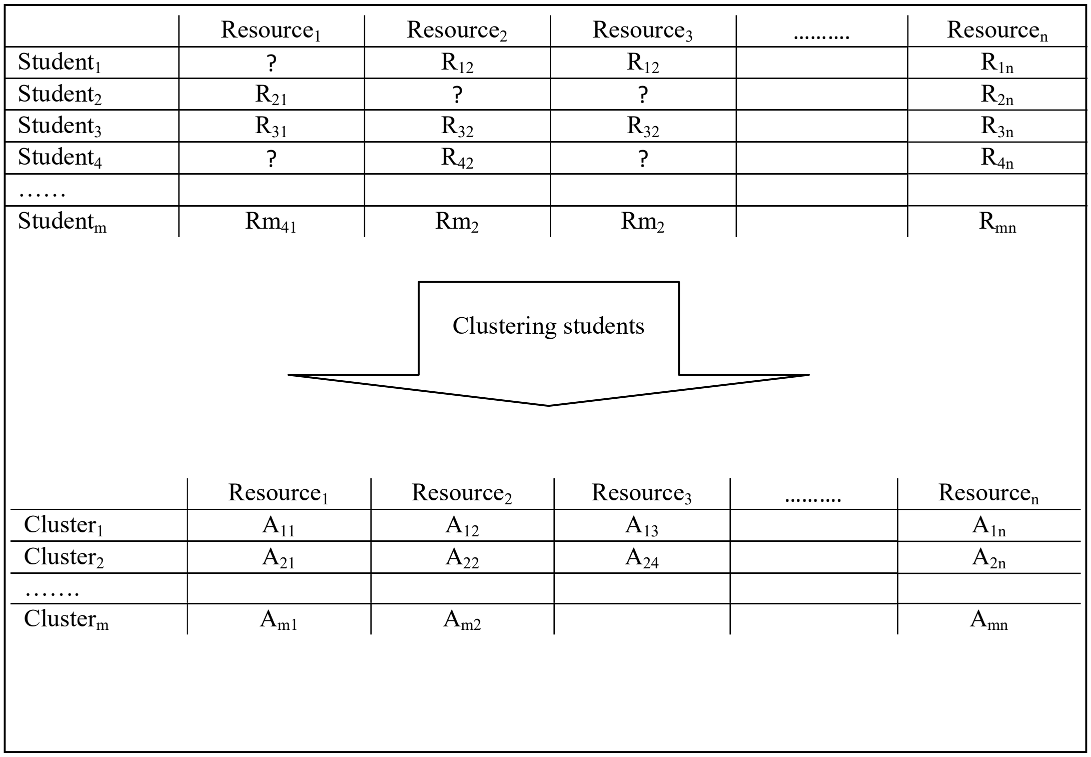
(Student actions along with the weights assigned to each action)

Web-based academic D-library needs to use Web data analysis tools to find the underlying usage patterns from usage data. Web usage mining techniques arise like an appropriate tool to explore useful patterns from users behaviours while interacting with a Web-based system (AGUILAR 2009; Esslimani, Brun et al. 2009; Umamaheswari and Srivatsa 2014). Xu noted that Web usage mining hadbeen proposed as a method for revealing user access patterns from usage data (Xu 2008). Academic D-library system has a wealth of log information resource, which makes it a good environment to apply Web usage mining techniques in order to build a rich knowledge base with a higher value for making decisions for recommendations (Zhu and Xu 2011).

3.2.2 Web Clustering Usage Mining

The sparsity problem has anegative impact on the recommendation quality of collaborative filtering systems. Thus it represents a challenge to the collaborative filter recommender systems (Gao, Xing et al. 2007; Gong 2010). Due to sparsity problem, the similarity between users in memory-based recommender methods is difficult to define (Manh Cuong Pham 2011). The authors (Ahn, Cho et al. 2004; Cho and Kim 2004; Su and Khoshgoftaar 2009; Kuzelewska 2014) mentioned that, building recommender system based models alleviate the sparsity. Accordingly, a clustering-based collaborative filtering framework is proposedto overcome sparsity and provide better recommendation result in terms of accuracy. The authors (Li and Kim 2003; Borhani-Fard 2013; Bargah and Mishra 2016) noted, clustering mining technique is often employed to address the sparsity problem. It has been proved that clustering mining provides better recommendation quality when used with collaborative filtering (Xue, Lin et al. 2005; Kim and Ahn 2008; Manh Cuong Pham 2011).

Web usage clustering mining plays a prominent role to predict patterns and infer rules, which helps for building the data prediction model. A prediction model is used for active students to predict unknown ratings better. The studies (Banerjee and Ghosh 2001; Cadez, Heckerman et al. 2003; Chen and Liu 2003; Pallis, Angelis et al. 2005; Pallis, Angelis et al. 2007) showed that Web data mining clustering is used for understanding users' navigation behaviour. In respect to the sparsity problem, the idea of using Web usage clustering is to reduce the dimensionality of the student–resource interaction matrix resulting partitions of users and use these partitions as neighbourhoods. Figure 3.1 explains how to address the problem of sparsity problem using a student-resource matrix that is very sparse. The figure illustrates the matrix before applying clustering and after applying clustering. In the above matrix (before clustering) row r_i represents the interests of student i and consists of a list of resources R_{ri} which indicates the student's interest in those resources. The first task is to divide the student-resource matrix into partitions,and the students are grouped based on their feedback data similarity. The output of this step is sub-matrices of the original matrix, where each matrix includes a set of similar students. Subsequently, a dense rating sub-matrices are gained. Density dataset can effectively address data sparsity problem (Lee 2015). When a target student arrived, predictions are made using these sub-matrices as basic units for the prediction process. The process of recommendation begins by assigning the active student to the cluster with the largest similarity by comparing the student's active profile with the cluster centres (cluster profile), and the prediction is computed based on the opinions of students in the same cluster.



	Resource$_1$	Resource$_2$	Resource$_3$	....	Resource$_n$
Student$_1$	?	R$_{12}$	R$_{12}$		R$_{1n}$
Student$_2$	R$_{21}$	?	?		R$_{2n}$
Student$_3$	R$_{31}$	R$_{32}$	R$_{32}$		R$_{3n}$
Student$_4$	?	R$_{42}$	?		R$_{4n}$
……					
Student$_m$	Rm$_{41}$	Rm$_2$	Rm$_2$		R$_{mn}$

	Resource$_1$	Resource$_2$	Resource$_3$	....	Resource$_n$
Cluster$_1$	A$_{11}$	A$_{12}$	A$_{13}$		A$_{1n}$
Cluster$_2$	A$_{21}$	A$_{22}$	A$_{24}$		A$_{2n}$
…….					
Cluster$_m$	A$_{m1}$	A$_{m2}$			A$_{mn}$

Fig. 3.1: Clustering student-resource matrix, where the row r_i represents the interests of cluster i and consists of a list of resources, A_{mn} indicates the cluster centres

The following example shows asparsity student–resource rating matrix for five students and eight resources in a resource recommender system.

Example 3.1: Given the m × n student–resource rating matrix includes 5X8 = 40 elements such that there are 5 students S = { St_1, St_2,......, St_5} and 8 resources R = { R_1, R_2, ..., R_8}. The student–resource rating matrix rows represent the students, and the columns represent the resources. The student interest on the resources is assigned by a value F where F is the set of student feedbacks, F = {1, 2, 3, 4, 5}. Otherwise, the student interest is 0, 0 indicate that the student has not rated a particularresource.

$$S = \left\{ \begin{array}{l} s \in \text{ F if student i rated resource j} \\ s = 0 \text{ Otherwise} \end{array} \right\}$$

We have 5 students and 8 resources, and ratings are integers ranging from 1 to 5, the matrix may look as shown in figure 3.2. As shown in Figure 3.2 most student-resource pairs have blanks, meaning that the students have rated a few resources among the total number of available resources.

Using denser data produce effective recommendation results (Lee 2015). Thus, in this example, the sparsity level of data sets factor is taken into consideration to measure the sparsity level before and after clustering. Equation 3.1 shows the computing of the sparsity level (Sarwar, Karypis et al. 2001).

$$\text{Sparsity level} = 1 - \text{nonzero entries} / \text{total entries} \qquad \text{Equation (3.1)}$$

The sparsity level of the original matrix (Figure 3.2) is:

$$1 - 24/40 = 0.4$$

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
St_1	1	3	0	3	4	0	0	0
St_2	2	0	0	0	4	0	5	1
St_3	0	4	0	4	4	2	3	5
St_4	0	4	0	2	0	4	2	5
St_5	3	0	0	3	4	0	4	5

Fig. 3.2: Original student-resource matrix

Given the student-resource rating matrix (figure 3.2), the k-means clustering technique is applied resulting in two sub-matrices as shown in figures 3.3, 3.4. The students are divided into 2 clusters: C1 ∈ {St_1, St_2, St_5} and C2 ∈ { St_3, St_4}. The original matrix high sparsity is transferred into a relatively low sparsity matrix as shown in figure 3.5.

	Cluster 1							
	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
St_1	1	3	0	3	4	0	0	0
St_2	2	0	0	0	4	0	5	1
St_5	3	0	0	3	4	0	4	5

Fig. 3.3: Cluster 1 - student-resource submatrix

	Cluster 2							
	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
St_3	0	4	0	4	4	2	3	5
St_4	0	4	0	2	0	4	2	5

Fig. 3.4: Cluster 2 - student-resource submatrix

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
Cluster 1	2	1	0	2	4	0	3	2
Cluster 2	0	4	0	3	2	3	2.5	5

Fig. 3.5: Cluster representatives (Student-resource interaction matrix after clustering)

The sparsity level of the student-resource interaction matrix after clustering is:

$$1 - 12/16 = 0.25$$

The sparsity level of the original matrix after clustering (= 0.25) is less than it was before clustering (= 0.4).

3.3 Web Usage Mining Framework

In this section, the proposed Web usage mining framework is described. Firstly, the overall system is described, followed by detailed design and construction methods.

3.3.1 Framework overview

In figure 3.6 the overall architecture of the Web usage mining framework is presented. The architecture consists of two main components: the off-line component - a data model based collaborative filtering and the online component - generation of recommendations. Each of these components is discussed briefly below.

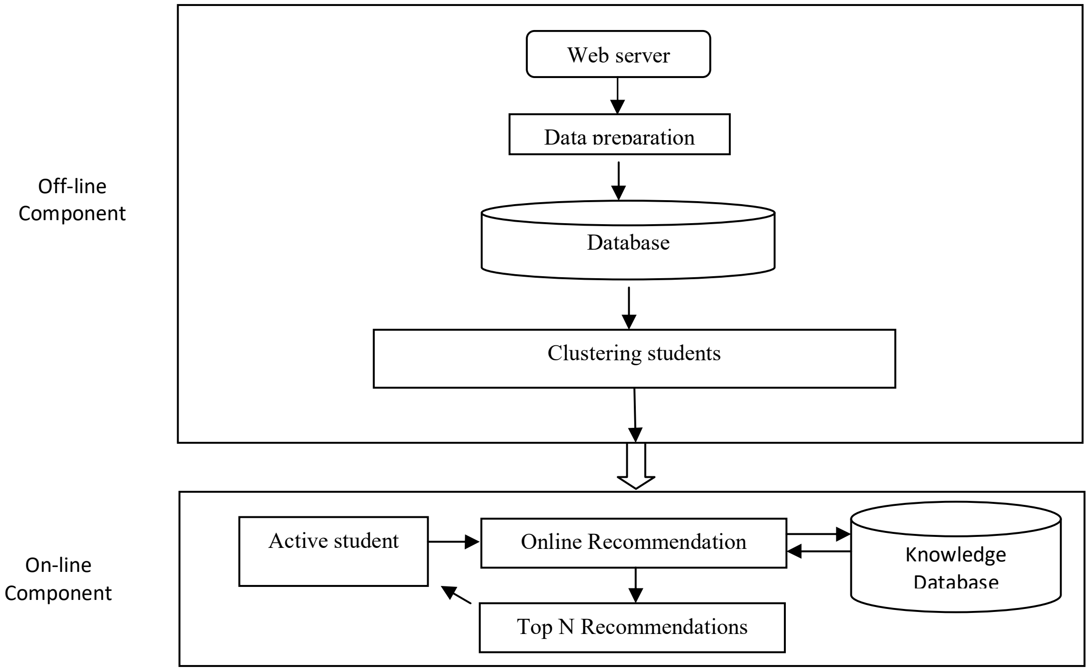


Fig. 3.6: Architecture of the Web Usage Mining Framework

The off-line component is responsible for acquiring the clusters profiles. It comprises two sequential stages: Data preparation and Data mining clustering stages. In the first stage, theweblog fileistransformed into transaction forms that are fit for data mining. The second stage is responsible for partitioning the students into clusters of similar behaviour using k-Means clustering algorithm. It is better to execute data preparation and data mining processes off-line since they take large amounts of processing time which do not match the requirement of real-time recommendation service. By the end of the component stages, a model based collaborative filtering is constructed. The model will be used by the online component to generate online recommendation results.

The on-line component is responsible for providing recommendations in real time to the active student based on the built model. After the representative clusters have been computed in the off-line component, the framework process is completed by the online component which begin by constructing the student profile on the basis of detailed observations of student's interaction with the D-library Website, then matching this profile against the cluster profiles to classify the active student to the best cluster profile represents their interests. The chosen cluster profile represents the source of selection whereby a recommendation list is to be generated.

The off-line and on-line components are interrelated to each other as follows:

a) The online component offers recommendations based on the clusters provided by the off-line component.

b) The off-line component generates correspondent rules with the data accumulated online.

Figure 3.7 shows the interactions between the off-line and online components on a high abstraction level. The input to the off-line component is the web server feedback data, and the output is the derivation of aggregate feedback profiles. The input to the online component is the active student profile, and the clusters generated by the off-line component, and the output will be a set of recommended resources.

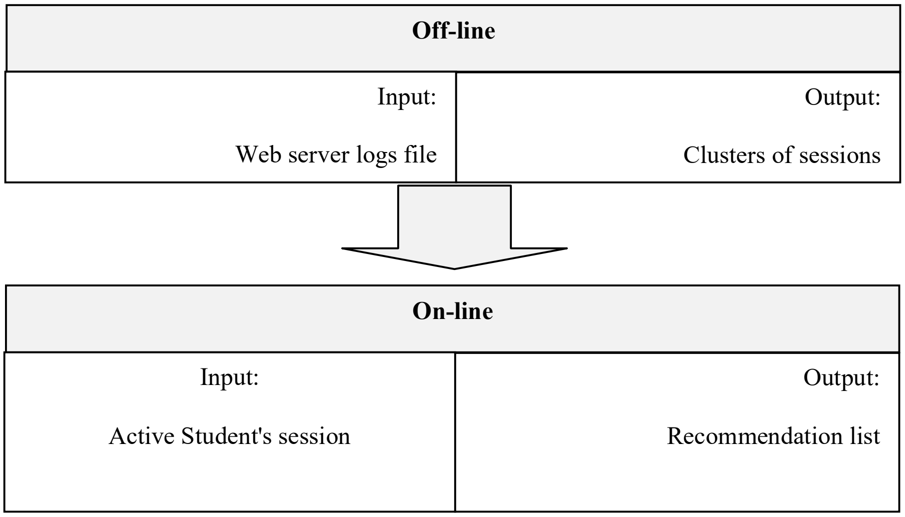


Fig. 3.7: Interrelation between off-line and On-line components

The system provides recommendations according to the following general steps:

(1) Clustering the students based on past similar feedback data.

(2) Once a student starts a session, they provide the system with feedback data which represents the target student's preferences.

(3) Cosine measure is used to calculate similarity values between the target student and the clusters' profiles to assigns the active student to the best cluster.

(4) Select the "nearest neighbours" of the target student in the cluster using Pearson correlation coefficient method.

(5) Finally, the system presents a set of Top-N resources.

3.3.2 Web Usage Mining Framework Architecture

The following sections will describe the Web usage mining framework components in details.

3.3.2.1 The Off-Line Component

This section describes the two stages taken by the off-line component, namely data preparation and clustering students. The off-line component processes are performed as illustrated in Figure 3.8.

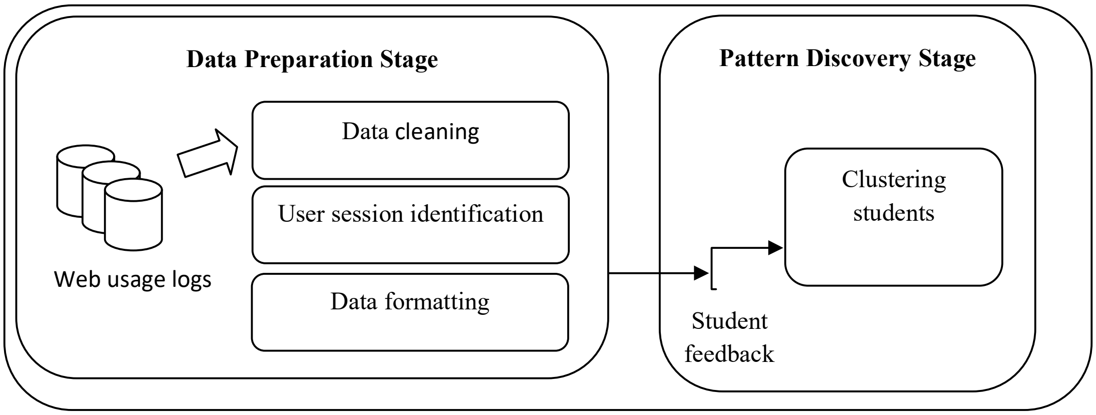


Fig. 3.8: Off-line component processes

Stage 1: Data Pre-Preparation

This stage contains two phases: data acquisition phase and pre-processing phase.

A. Data acquisition phase

Pre-preparing data stage begins first by collecting necessary data. The data involved in the academic D-library system is massive mostly it contains users information, resources information, security information, borrowing information etc, as well as usage web data. In the context of this study, the proposed Web recommendation will be based on Students' usage log data. Whenever the student interacts with the D-library website contents the interaction details are recorded in the web server in the form of web log files; these log files provide the system with the navigational behaviour knowledge of students in terms of access patterns.

B. Data pre-processing phase

The data pre-processing process is considered as the important activity in web usage mining. This study considers Web log file as the main source of information to capture the students' access trends. Weblog data are maintained in the web servers in the form of plain text files (Grace1, V.Maheswari et al. 2011). Since these data have not been intended a priori to serve for data mining processes, application of pre-processing steps have to be applied to Weblog data to preserve the type of information required for data mining processes (Wahab, Mohd et al. 2008; Ramakrishna, Gowdar et al. 2010). These pre-processing tasks are the same for any web usage mining problem and discussed by Cooleyet al (Cooley, Mobasher et al. 1999). In this phase, Web log files are transformed into transaction data that are required for the mining process. The data pre-processing process in this study contains three sequential steps: Data cleansing, user session identification, and data formatting. The pre-processing steps are shown in Figure 3.9.

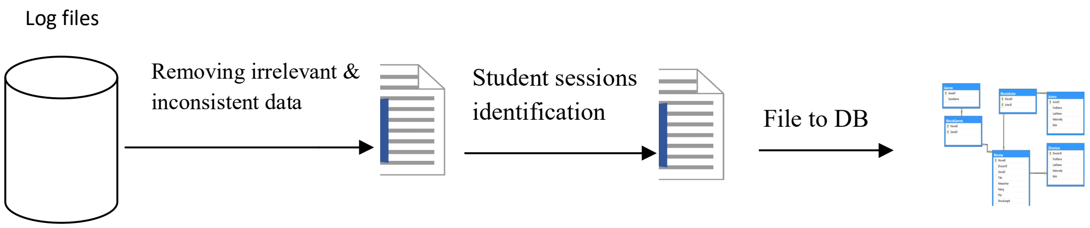


Fig. 3.9: Pre-processing steps

Data cleansing: In this step, irrelevant and noisy data (images, graphics, Multimedia etc.) are removed from the log files,and a minimized log file is obtained.

User session identification: The session gives a complete set of activities done in the D-library contents by the user in aspecific time period. In case of web users, it is not trivial which actions belong to which user (Iváncsy and Juhász 2007). Thus, it is important to distinguish between different students for analysing different student access behaviour patterns. Since the academic D-library Web site is accessed by registers students, the system uses the "university ID number" attribute to identify each student session. A session ends when a student logs out.

Data formatting: Data formatting is the process that converts the log data intoa standard format that are suitable for data mining (Dong, Nie et al. 2006). In order to make use of students' activities, a conversion method is proposed. The conversion method comprises the following two steps:

1. Building resource access table.
2. Converting the student activities to numerical values.

Building resource access table

In this system, students do not need to rate resources explicitly to perform their preferences, so the system needs to use implicit ratings. Within the D-library environment, implicit interest on a particular resource is represented by five actions. These actions indicate some kind of interest in the resource during a session. The action set can be defined: A= {printing a resource, downloading a resource, bookmarking a resource, reading a resource, viewing a resource's abstract}. This action set reflects how much the resource has been interesting to the student.

In the D-library system, each student accesses the resources among the available resources. These accessed resources can be displayed in the form of a table. Given n student sessions as S = {s_1, s_2, …. s_n} and m resources as R = {r_1, r_2, ….r_m} stored in a Web log file, we built up the resource access table. The resource access table reflects the resources accessed by students in various sessions and what actions the students perform on it. There may be more than one action performed by a student for a particular resource in a session. For building the resource access table no need to save less action for a resource, the highest action for a resource only remains and all other actions on that resource in a session are removed, e.g. if a student views the abstract of a resource, then views its contents, and then downloads the resource, in this case all action rating are discarded but the highest rating for the resource will be used in the usage matrix. As an example of resource access table, consider figure 3.10 which shows a resource access table. The row i represents the interests resources accessed by the student during a session, and the columns of the table represent the type of access done on a particular resource. The resource access table cells illustrate the action performed by a student on a resource where P represents printing the resource, D represent downloading the resource, B represent bookmarking the resource,R represents reading the resource, and A represent viewing a resource abstract. These actions are represented as a binary representation, i.e. 1 if the student performed the action on a resource, 0 if there is no action performed by the student on a resource.

Resources	R_1					R_2					………	R_n				
Action Students	P	D	B	R	A	P	D	B	R	A		P	D	B	R	A
Student$_1$					1	1									1	
Student$_1$			1						1					1		
….																
Student$_n$				1				1							1	

Fig. 3.10: Resource access table

Convert Student activities to numerical values

After building the resource access table, each action must have a score. This step converts the resource access table into student-resource interaction matrix by using the actions weightings table (table 3.1) mentioned in section 3.2.1.

Consider a student's session S. Each session action is represented by a pair of number (R_i, A_j), where Ri represents the resource code, and A_j (j=1, 2, 3, 4, 5) the student action.

A session file is created S : {(12,3), (6,1), (15,2),……} According to table 8.1, S shows that a particular student downloaded resource 12 and only viewed the abstract of resource 6 and viewed resource 15 in the screen, etc.

We need to express the student behaviour by a feature vector of numeric ratings on various resources to apply the similarity functions which are used with collaborative filtering systems. For this purpose,the student's activities in the resource access table (figure 3.10) is converted to numerical values according to table 3.1 to build the matrix (figure 3.11). We call this matrix the student-resource matrix. This matrix reflects the level of student's interest for resources accessed by them. The student-resource interaction matrix is represented by a |I|×|J| matrix as shown in figure 3.11, such that the row i represents the resources accessed by a student during a session. Each row is expressed as a sequence of weight-resource pairs, $s_{i=}\{(r_1\ a_{i1}), (r_2\ a_{i2}),\ .....\ (r_j a_{ij})\}$, where a_{ij} stands for the weight on the resource j.

Resources Student	R_1	R_2		R_m
$Student_1$	1	5		2
$Student_2$	3	2		3
….				
$Student_n$	2	3		2

Fig. 3.11: student-resource interaction matrix

Stage 2: Clustering students

After the pre-processing procedure for Weblog is finished, it is still not ready for effective application in collaborative filtering recommendation because the number of transactions is very huge. It will not be feasible to deal with huge number of transactions because it consumes large amounts of processing time. A better solution would be to group the students into clusters, with each cluster having similar rating. In this stage, a data clustering mining algorithm is performed to cluster the student-resource interaction matrix performed in the pre-preparation stage. The clustering mining algorithm generates for each cluster a profile cluster which demonstrates the most common students' preferences in the cluster. The goal of clustering students is to reduce the sparsity problem and to increase recommendations accuracy. This step also improves the system performance since the amount of data that must be analysed is much smaller (Sarwar, Konstan et al. 2002).

A variety of clustering techniques can be used for clustering students. This study employs k-means clustering algorithm to obtain students clusters. K-means is an unsupervised learning algorithm. The K-means algorithm is wide used clustering algorithm to cluster user transactions (Patel and Mehta 2011; Virmani, Shweta Taneja et al. 2015). The main reason for choosing k-means over other clustering algorithms is that it has the ability to handle high dimensional data such as those present in academic D-libraries and its low time complexity (Kuzelewska 2014; Wang, Yu et al. 2014). In addition, it is the most important clustering algorithm in recommender systems (Amatriain, Jaimes et al. 2011).

The working process of K-means algorithm is outlined as follows:

1. Randomly choose 'c' initial cluster centres (representing the K transaction groups) as the initial clustering centres.

2. Calculate the distance between each object (representing the transactions) and cluster centres. This is done by using Euclidean Distance metric to calculate the distance between cluster centroid to each object. Euclidean Distance is one of the most popular distance measures – it calculates the root of square differences between the coordinates of the points in each objects (Kouser and Sunita 2013). This can be written as:

$$D_{xy} = \sqrt{\sum_{k=0}^{n} (\mathrm{x}_{\mathrm{ik}} - \mathrm{y}_{\mathrm{jk}})^2} \qquad \text{Equation (3.2)}$$

3. Assign the objects (representing the transactions) to their most similar clusters (measuring from the cluster centre) according to their similarity (distance) with these clustering centres (i.e. The closest cluster centre to the object is the cluster centre with the minimum distance to the object).
4. Recalculate the centre of each cluster as the centroid of all the data points in each cluster. The new centre can be found by taking the average rating over all users for every resource assigned to the cluster. This can be written as:

$$\mathrm{c}_{\mathrm{i}} = \frac{1}{|Si|} \sum_{x_i \in S_i} x_i$$

   S_{i} is the set of all data points assigned to the i^{th} cluster

5. If the new centres are different from the previous ones, repeat Step 2, 3 and 4 until convergence of standard measure function appears. Otherwise, terminate the algorithm.

Figure 3.12 shows the base of the clustering algorithm used to cluster the students (**Algorithm 1**). The input of the algorithm is the student-resource matrix, and the output will be sub-matrices profiles (clusters centres) made by the average of the users' feedback data.

Algorithm 1: User-resource matrix clustering algorithm

```
Input:   Student-resource matrix;
         k, Number of clusters;
         Set of users $U = \{u_1,u_2, ..., u_n\}$;
         Set of resource $R = \{r_1,r_2, ..., r_m\}$
Output: Set of clusters centres  $\{ c_{11}, c_{12}, \cdot \cdot \cdot c_{kn} \}$
Begin
Random cluster partition set $CU = \{cu_1, cu_2, ..... , cu_n\}$
Cluster centroids set $C = \{c_{11}, c_{12}, .., c_{kn}\}$                    // The set of initial means
Initialize: z = 0; dis = 0; Mdis = 10;  NCluster = 0;
 Do
    for i=1, i = n, i++ do                                  // n number of users
        for y=1, y = k, y++ do                              // k  number of clusters
            for j=1, j = m, j++ do                          // m number of resources
                if $r_{ij}$ Not Null and $c_{yj}$ Not Null then  // Only resource rated by the users and –
                                                            // cluster centroids are used for clustering
                    $z = z + (r_{ij} - c_{yj})^2$                  // Calculate the distance between users $r_{ij}$ –
                                                            // and cluster centroid $c_{yj}$ using Eq. 3.2
            end for
            dis = Sqrt(z)
            if dis < Mdis then
               Mdis = dis
               NCluster = y                                 // Number of nearest cluster
            end if
        end for
       Assign $u_i$ to $cu_{NCluster}$                              // Assign users to clusters with -
                                                            // minimum distance Mdis
    end for
                                                            // Find the new clusters centroids
    for i=1, i = k,  i++ do                                 // k  number of clusters
        for j=1, j = m, j++ do                              // m number of resources
          $CUMean^{new} = \frac{1}{number( users \in CU)} \sum_{x_j \in S_j} x_j$     // Compute the average value of the –
                                                            //assigned points S in each cluster
          change = true
          if $c_{ij} \neq$ $CMean^{new}$ then
             $c_{ij} = CMean^{new}$                        // Update centroid $c_{ij}$ value to new centroid location
          else
             change = false
          end if
       end for
 while change = true
  return $\{ c_{11}, c_{12}, c_{13}, ..., c_{kn}\}$
End
```

Fig 3.12: Clustering student-resource matrix algorithm

3.3.2.2 The Online Component

The online component is the main component of the proposed framework. It includes two models: Student profile manager model and Recommendation model. The following subsections will describe these models.

3.3.2.2.1 Student Profile ManagerModel

Recommender systems need user profiles in order to provide recommendations to them (Kveton and Berkovsky 2015). The user profile contains information about a user, enabling recommender systems to recommend items based on this information. Successful recommendations mainly depend on accurate users' profiles (Gauch, Speretta et al. 2007; Tejeda-Lorente, Porcel et al. 2014).

The student profile manager model is based on three factors:

a) Several different representation schemas for building user profiles were identified by Montaner et al. (Montaner, L\ et al. 2003) and Gauch et al. (Gauch, Speretta et al. 2007). The choice of the right schema depends on which recommendation technique will be used. The history-based model was chosen in this model to build students' profiles. The history-based model mainly focuses on the past users' activities (Lemire, Boley et al. 2005).

b) The model considers a short-term history of access behaviour to build the student's profile. The very earlier resources that the student accessed are less likely to affect the recommendation process. Thus considering a short-term history for making the decision about what to recommend is a good decision. The system restricts the short-term model to the n most resources recently accessed by the student and is termed as the active session (set to 3 in the model).

c) The model builds student's profile from the active student session. The active student session includes implicit feedback data. The implicit feedback data is transferred to numerical values according to the action weight table – table 3.1 mentioned in section 3.1.1.

According to these factors, the student profile is defined by a rating vector: S_u={$r_{u,1}$,$r_{u,2}$, $r_{u,3}$}. e.g. Consider an active student S downloaded $Resource_1$, printed $Resource_2$, and bookmarked $Resource_3$. The active student's profile is {$Resource_1$=4, $Resource_2$=5, $Resource_3$=3}.

The student profile manager model constructs student profile following two main steps as shown in figure 3.13. In the first step, the system monitors the active student's activities,and in the second step, the active student's activities are converted to numerical values representing the active student's profile.

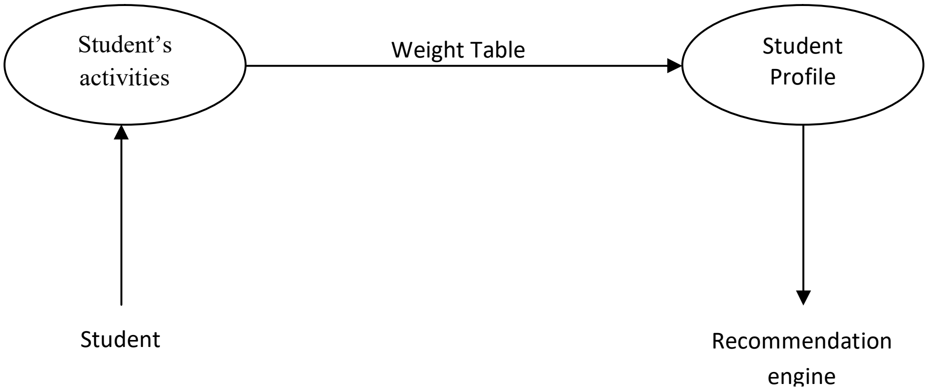


Fig. 3.13: Student profile construction

3.3.2.2.2 Recommendation Model

Its task is to generate a recommendation set for the current student session. The Recommendation model involves three phases as shown in figure 3.14. Since the similarity measure between users is the key issue in the collaborative recommendation algorithms, the first two phases are used to measure the similarity between active student and other students.

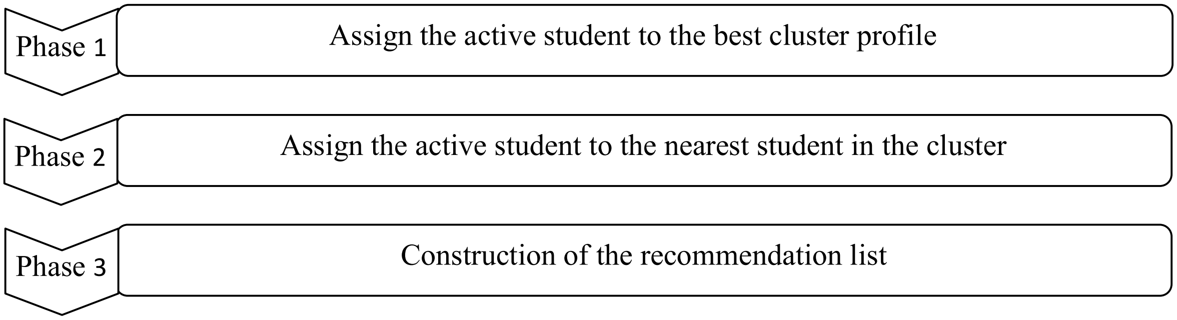

Phase 1
Assign the active student to the best cluster profile
Phase 2
Assign the active student to the nearest student in the cluster
Phase 3
Construction of the recommendation list

Fig. 3.14: Recommendation model phases

The process of the Recommendation model is described as follows:

[Input]: An active student profileandthe clusters' profiles.

[Output]: A set of resources in a descending orderof the predicted ratings.

Step 1: Assign the active student profile to the best usage cluster profile by matching the current student profile with the discovered usage cluster profiles.

Step 2: Find the top 10 nearest neighbours of the target student in the selected cluster.

Step 3: Calculate the prediction rating for the resources most liked by the top nearest neighbours computed by simple weighted average.

Step 4: Sort the resources in a descending order based on scores

Step 5: Select the top-N resources as the recommendation list.

Figure 3.15 illustrate the recommendation model process:

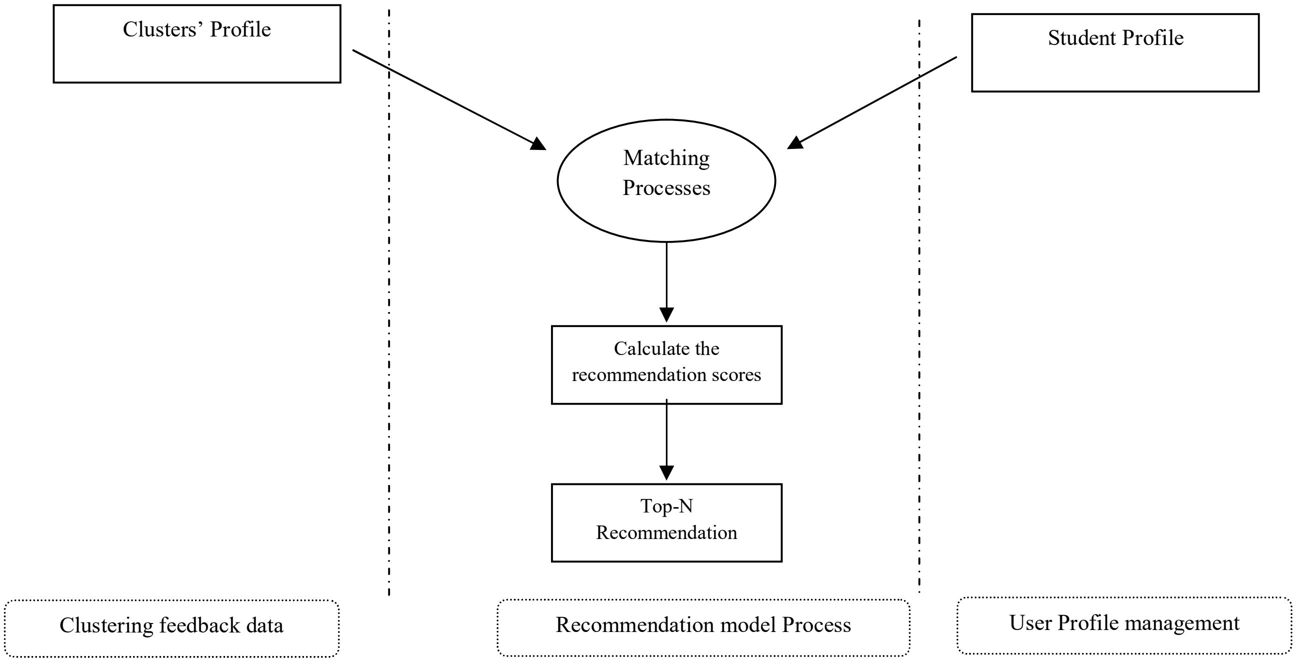


Fig 3.15: Recommendation model Process

The following subsections describe the Recommendation Model phases in details.

Phase1: Assign the current student to the best cluster profile

The goal of this phase is to match, at each step, the active student session with the aggregate cluster profiles generated in the offline component to select the best cluster profile. Cosine measure metric is used to compute similarity values between the target student and the clusters centres. The cosine measure metric is a measure of similarity between two vectors, with values between 0 and 1. A larger value is an evidence of high similarity. The similarity between user X and cluster Y is computed using the items which have been rated by user X and rated in the cluster profile. Equation 3.2 shows the cosine measure metric that calculates the similarity between user X and cluster Y.

$$Cosine(x,y) = \frac{\sum_{k=1}^{n} \mathrm{R_{xk}}\,\mathrm{R_{yk}}}{\sqrt{\sum_{k=1}^{n} \mathrm{R_{xk}}^{2}}\sqrt{\sum_{k=1}^{n} \mathrm{R_{yk}}^{2}}} \qquad \text{Equation (3.3)}$$

In this equation R_{xk} indicates the rating of the resource k by student X, R_{yk} is the rating of the resource k in the cluster profile Y,and n is the number of items co-rated by both students.

Figure 3.16 shows the algorithm (**Algorithm 2**) used in classifying the active student to the best cluster.

Algorithm 2: Classifying the active user to the closest cluster algorithm

```
Input:   Set of clusters CU = {$cu_1$, $cu_2$, ....... $cu_k$};
         Set of clusters centroids C = {$c_{11}$, $c_{12}$, ... $c_{kn}$};
         Active user profile, R= {$r_1$, $r_2$, ..., $r_m$}
Output: The user cluster
Begin
    for i=1, i = k, i++ do                     // Compute the similarity between active user and -
                                               // clusters centroids C using cosine metric Eq. 3.3
      Initialize: x = 0; y = 0; z = 0; sim = 0; CCluster = 0;
      for j=1, j = m, j++ do
        if $r_j$ Not Null and $c_{ij}$ Not Null then    // Only resource rated by the active user and rated in –
                                               // the cluster profile are used in Eq.3.3
          $x = x + (r_j * c_{ij})$
          $y = y + (r_j * r_j)$
          $z = z + (c_{ij} * c_{ij})$
        end if
      end for
        $s_i = \frac{x}{Sqrt(y) * Sqrt(z)}$
        if $s_i$> sim then
          sim = $s_i$
          CCluster = i                         // Number of closest cluster
        end if
    end for
  return $cu_{CCluster}$                          // The closest cluster to the active user is chosen
End
```

Fig. 3.16: Assigning the active student to the best cluster profile algorithm

Phase 2: Select the nearest students to the active studentin the cluster

After assigning the active student to the best cluster, next step is to select the most similar students to the active student. This is called the nearest neighbour approach (Freund, Iyer et al. 2003). This approach results in more accurate predictions since the recommendations are predicted using the ratings of neighbours (Herlocker 2002; Ekstrand, Riedl et al. 2011). This phase involves two steps: in the first step, the similarity between the active student session and students in the cluster is calculated, in the second step the k-top student's neighbourhood are selected.

First step: Similarity computation

The basic idea of similarity computation in collaboration filtering algorithms is to identify the similar users to the active user in terms of the rating patterns. Many approaches used for this propose include the vector similarity-based approach (Breese, Heckerman et al. 1998), the Spearman rank correlation, the Pearson-Correlation based approach (Sun, Kong et al. 2005) and the extended generalized vector-space model (Soboroff and Nicholas 2000). In this study, Pearson correlation coefficient method is usedto compute similarity values between the target student session and the students in the cluster. Pearson correlation coefficient approach provides better quality in collaborative filtering than other approaches (Shenoy, Jain1 et al. 2013). The Pearson correlation coefficient measures the strength of a linear relationship between two vectors of ratings using a value between -1 and +1. A positive value is the evidence of a positive linear correlation where both users move in the same direction of rating,i.e. high ratings of user X are associated with high ratings of Y,and low ratings of X tend to be associated with low ratings of Y; A negative value is the evidence of negative linear correlation; A value of 0 denotes that there is no relationship between the two users. Equation 3.3 shows the Pearson correlation coefficient.

$$Sim(x,y) = \frac{\sum_{i \in I_{xy}} (r_{x,i} - \bar{r}_x)(r_{y,i} - \bar{r}_y)}{\sqrt{\sum_{i \in I_{xy}} (r_{x,i} - \bar{r}_x)^2} \sqrt{\sum_{i \in I_{xy}} (r_{y,i} - \bar{r}_y)^2}} \qquad \text{Equation (3.4)}$$

In this equation I_{xy} indicates the set of items rated by both user X and user Y, $r_{x,i}$ is the rating of user X on item i, $r_{y,i}$ is the rating of user Y on item i and $\bar{r}_x$, $\bar{r}_y$ are user's average ratings of the i-th item.

Second step: Neighbourhood Selection

After computing the similarity between the target student session and the students in the cluster, the next task is to select the most similar students for the active student. This is an important task since the recommendation quality depends on the ratings of neighbours (Manh Cuong Pham 2011). These nearest neighbours serve as a source of prediction whereby a recommendation list is to be generated from those resources preferred by active student’s neighbours, meaning that generation of recommendations will be based on a much smaller number of sessions than in the whole cluster. There are two strategies for selecting neighbours: the top-N strategy and threshold-based strategy (Dakhel and Mahdavi 2011). Threshold-based strategy selects the users whose similarity exceeds a certain threshold value, in contrast, the top-N strategy select the most similar neighbours purely according to their similarities with the active user. In this study, the two strategies are followed. Figure 3.17 shows the nearest users selection algorithm (**Algorithm 3**).

Algorithm 3: Select the most similar users algorithm

```
Input:   Active user profile, $R=\{r_1, r_2, .., r_k\}$;
         Active user ratings, $RV=\{r_1^{Rate}, r_2^{Rate}, .., r_k^{Rate}\}$;
         Set  of cluster users $U=\{u_1, u_2, .., u_n\}$;
         Set of resources rated by cluster users $UR=\{ur_{11}, ur_{12}, .., ur_{nm}\}$;
         Cluster users ratings $URV = \{ur_{11}^{Rate}, ur_{12}^{Rate}, .., ur_{nm}^{Rate}\}$;
         threshold value, thv; Top-N value, topv
Output:  Set of nearest users $NU = \{ nu_1, nu_2, \cdot\cdot\cdot nu_t\}$
Begin
    for $i=1, i <= number(u \in U), i++$ do          // Compute the similarity between active user and –
                                                     // sub-matrix users using Pearson method Eq. 3.4
        Initialize: $x = 0; y = 0; z = 0; sumr = 0; sumur = 0;$
        for $j=1, j <= number(r \in R), j++$ do
            for $k=1, k <= number(ur_j \in UR), k++$ do
                if  $r_j == ur_{ik}$and $ur_{ik}^{Rate}$ Not Null then  // Only resource rated by both the active user –
                                                     // and users in the cluster are used in Eq. 3.4
                    $c = c+1$                        // Count number of resources co-rated by both users
                    $sumr = sumr + r_i^{Rate}$
                    $sumur = sumur + ur_{ik}^{Rate}$
                    break for
                end if
            end for
        end for
        $avgr = \frac{sumr}{c}$                      // Computing the active user's average ratings
        $avgur = \frac{sumur}{c}$                    // Computing the user average ratings
        for $j=1, j <= number (r \in R), j++$ do
            for $k=1, k <= number (ur_j \in UR), k++$ do
                if  $r_j == ur_{ik}$ and $ur_{ik}^{Rate}$ Not Null then
                    $x = x + ((r_j^{Rate} - avgr) * (ur_{ik}^{Rate} - avgur))$
                    $y = y + ((r_j^{Rate} - avgr) * (r_j^{Rate} - avgr))$
                    $z = z + ((ur_{ik}^{Rate} - avgur) * (ur_{ik}^{Rate} - avgur))$
                end if
            end for
        end for
        $ps_i = \frac{x}{Sqrt(y)*Sqrt(z)}$           // Computing users' Pearson similarity value
        if $ps_j > thv$ then                         // Selecting the most nearest users according to a –
                                                     // threshold value, thv
            Add $u_i$to the set of top nearest-neighbours NU
        end if
    end for
    order users NU according to their $ps_i$         // Updating the set of top nearest-neighbours
                                                     // according to the Top-N value, topv
    return $\{nu_1, nu_2, ..., nu_{topv}\}$
End
```

Fig. 3.17: Nearest users selection algorithm

Phase 3: Construction of the recommendation list

A recommender system should provide ranking recommendations to the students to minimize the effort required from them to find highly relevant resources. Ranking resources is a beneficial function for students, i.e. better decision-making, time-saving, less resource search. The goal of this phase is to ultimately derive the top-N recommendation that provides the most "value" to the active student. In doing so, this phase provides the top-N resources in two steps: first aggregate the ratings of the top nearest-neighbours to generate predictions, then providing the recommendation list.

First step: Generating prediction

User-based collaborative filtering generatespredictions for users based on previously rating from similar users. In this study, weighted sum approach (Schafer, Frankowski et al. 2007) is used to generate predictions. The weighted sum approach is not the only method that has been used for computing predictions (i.e. multivariate regression method, weighted average approach) but it is the most common method used in collaborative filter applications,and it produces consistent results with models of human behaviour (Adomavicius and Tuzhilin 2005). The weighted sum method calculates a rating for an item using all the ratings of the neighbours on that item. Equation 3.5 shows the weighted sum method. The equation assigns a weight for each input rating to weight ratings from students who are most similar to the active student (selected by the previous step) in calculating the overall rating. The equation considers both positive feedback "value from 1 to 4" and negative feedback "1". Using negative feedback help avoiding recommending non-relevant resources, as the number of neighbours rated negative feedback for a resource increase, the overall weight for the resource gets smaller, thereby avoiding recommending non-relevant resources.

$$P_{x,i} = k \sum_{y \in S} \left| Sim(x,y) \right| . r_{yi} \qquad \text{Equation (3.5)}$$

Where $P_{x,i}$ is the prediction for the active student x for resource i, S is the top nearest-neighbours to the active student, the summations are all the students y ∈ S who have rated resource i, Sim(x,y) is a function that calculates the degree of similarity between users x and y to represent the similarity weight between user x and user y, and k serves as a normalizing factor and is usually defined as follows (Adomavicius and Tuzhilin 2005; Schafer, Frankowski et al. 2007) :

$$K = 1 / \sum_{y \in S} |sim(x, y)|$$

Second step: Providing recommendation list

This step provides a ranked list of resources in descending order based on prediction ratings,i.e. Top-N recommendation. The list includes resources not accessed by the active student. Figure 3.18 shows the construction of the recommendation list algorithm (**Algorithm 4**).

```
Algorithm 4: Construction of the recommendation list algorithm
Input:   Set of top nearest-neighbours, TU = {$tu_1, tu_2, …, tu_n$};
         Set of resources rated by nearest-neighbours, NUR = {$nur_{11}, nur_{12}, …, nur_{nm}$}ss;
         Top nearest-neighbours ratings NURV = {$nur_{11}^{Rate}, nur_{12}^{Rate}, .., nur_{nm}^{Rate}$};
         Pearson similarity values for nearest-neighbours, PS = {$ps_1, ps_1, …, ps_k$}
Output: List of recommendations R = {$r_1, r_2, \cdots r_n$},
Begin
      DNUR[ ] = DISTINCT (NUR[ ])
     for i=1, i <= number(dnur ∈ DNUR), i++ do
         for j=1, j <= number(tu ∈ TU), j++ do
             for k=1, k <= number($nur_j$∈ NRV), k++ do
                 if  $nur_{ik}$ == $dnur_i$then
                     $p_i = \frac{1}{ps_j} * p_i + (ps_j * nrv_{jk}^{Rate})$      // Compute the weight for each resource rated –
                                                                 // by top nearest-neighbours Eq. 3.5
                 break for
                 end if
          end for
      end for
   return {$r_1, r_2, \cdots r_k$ }
End
```

Fig. 3.18: Construction of the recommendation list algorithm

3.4 Summary

This chapterhas described the proposed Web usage mining framework. The framework incorporates usage data with the clustering data mining technique in the recommendation process as a means to help reduce the sparsity problem. The advantages of using the usage data and clustering are shown in this chapter. Then, the proposed framework is described. The proposed framework is divided into two main components: off-line and online components. The off-line component is comprised of two stages: data pre-processing and the derivation of student clusters. The online component is comprised of two stages: building student's profile and generating recommendations. The second stage consists of three steps, in the first step, the target student profile is classified to the closest cluster profile. In the second phase, the most similar students are selected and used as a source for generating the recommendation. Finally, a list of recommendations is presented.

CHAPTER 4

EVALUATION

4.1 Introduction

In the previous chapter, the proposed framework is described, and the usefulness of the techniques proposed to reduce the sparsity problem is presented.

The main focus of this chapter is to determine if using clustering technique and implicit feedback data lead to more accurate recommendation results compared to the memory-based collaborative method. In section 4.2 the evaluation tasks are presented. Section 4.3 presents the experimental setup. Section 4.4 experimental results are presented. In section 4.5 the experimental results are discussed. Finally, in section 4.6 the summary of this chapter is given.

4.2 Evaluation Tasks

The research is based on the suggestion that using clustering technique and implicit feedback data will improve the recommendations. In order to carry out the evaluation of the proposed recommender framework the following two tasks are performed:

First task: Evaluates quality of recommendations generated by using clustering explicit feedback data

To complete this task, ten experiments are conducted with a different number of clusters. The clustering technique is applied on the explicit feedback data. Then, the quality of the recommendations generated by the experiments is compared with recommendations generated by the standard K Nearest Neighbours (KNN) method. K Nearest Neighbours is a memory-based collaborative method, and it is usually used as a reference procedure. For K Nearest Neighbours the Pearson nearest neighbours algorithm is used. The Pearson nearest neighbours algorithm is one of the most commonly memory-based collaborative filtering algorithms (Chen, Wu et al. 2011), in which the recommendations are provided based on the neighbours of the target user from the entire database.

Second task: Evaluates quality of recommendations generated by using the proposed framework

To complete this task, implicit feedback data are added to the experiments conducted in the first task. Then, the quality of recommendations generated by these experiments is compared with recommendations generated by the standard K Nearest Neighbours (KNN) algorithm, and recommendations generated by using clustering only explicit feedback data.

4.3 Experimental Setup

Evaluation metric Different evaluation metrics can be used to measure the quality of collaborative filtering systems. In this study, the Mean Absolute Error (MAE) is used. It is a statistical accuracy metrics; statistical accuracy metrics measure the deviation of recommendation results from their real user ratings. MAE metric is the most commonly used metric to measure the accuracy of recommendations. MAE computes the average of the absolute difference between predicted ratings and real ratings that are actually assigned by users (Park 2013), this measure is formulated as shown in equation (4.1).

$$\text{MAE} = \frac{\sum_{i=1}^{N}|R_{ui} - \widetilde{R_{ui}}|}{N} \qquad \text{Equation} \quad (4.1)$$

Where N is the number of items, and R_{ui} represent the rating given to item i by the user u, $\widetilde{R_{ui}}$ represent the predicted rating for user uon item. Lower MAE is, the more accurately prediction.

Data sets The “Book-Crossing” dataset (http://www.informatik.uni-freiburg.de/~cziegler/BX/) is used as the data source for the experiments, it is one of the benchmarks datasets that are used to evaluate recommendation algorithms. It is collected from the book sharing site bookcrossing.com in August-September 2004. Bookcrossing.com is a well-known resource sharing site. This dataset is chosen for evaluating the proposed framework because it has high sparsity level and it contains implicit rating. The Book-Crossing dataset is composed of the following three tables:

1. BX-Books: contains the books identified by the ISBN, title, author, year of publication, publisher URL.
2. BX-Users: contains user IDs, location and age when available.
3. BX-Book-Ratings: contains the user ID, ISBN, book ratings. The BX-Book-Ratings table contains 278,858 users providing 1,149,780 integer ratings (explicit / implicit) for about 271,379 books. Book rating scale from 0-10, the explicit ratings on a scale from 1- 10 and the rate Zero indicate implicit rate.

Obtaining numerical values for the implicit rating In Bookcrossing.com, users can assign resources integer ratings from 1 to 10 (explicit rate). If the user accessed a book without rating it, the system would record zero (implicit rate) as a rate for this book. Book-crossing dataset do not specify a description of the implicit feedback (level of interest), to address this situation, numerical values for different levels of interest were obtained by using the books which are rated implicitly in the sub-matrix (clusters) by the user. Since users in the same cluster have the same interest, we assume that the book rated by a user can have the same ratings by other users in the cluster. Thus numerical values for different levels of interest for the books (only books which come under the target user interest - rated implicitly) were obtained by using the average of the users' ratings of the book in the cluster. After obtaining numerical values for the implicit rating, these values are added to the sub-matrices for users who had rated these book simplicity. For a review of the source code, see (Appendix C)

Limiting the dataset The experiments were performed on a subset of the Book-Rating table.The noisy data can adversely affect the results of the clustering data mining process. Thus, the data set is limited in two ways. First, the users that have rated less than 15 resources were removed. Second, the resources that are rated by less than 2 users were removed. After limiting the dataset, the remainder dataset constituted 131,778 records (explicit ratings) and 292,651 (implicit rating) from 4,911 users and 18,769 books. This data is loaded into the Microsoft Access database, and all experiments were conducted using C#.

Target users To evaluate the proposed framework sample 10 users data is tested, TUser = {100459, 8067, 97874, 60707, 31556, 75591, 7346, 60244, 100906, 76499}. For each user, 50% of their feedback data is used as testing data and the rest of their feedback ratings used as training data, i.e. each test user is removed from the whole dataset and then compared to generated recommendations.

Table 4.1 shows the selected target users for the experiments and their number of rated resources.

Target user ID	TU_1 100459	TU_2 8067	TU_3 97874	TU_4 60707	TU_5 31556	TU_6 75591	TU_7 7346	TU_8 60244	TU_9 100906	TU_{10} 76499
No. of rated resources	81	90	85	105	110	144	227	294	240	221

Table 4.1: Target users' number of rated books

User-resource matrix The dataset is represented as a user rating matrix, whose rows correspond to a particular user and whose columns correspond to a particular resource, and where each cell corresponds to the score of the user for a resource. The user-resource matrix had 4,911 rows and 18,769 columns with 131,778 ratings. Therefore, the user-resource matrix sparsity level is 1-((131,778)/ 4,911 * 18,769), which is 99.86%.

Dataset rating The cross-booking dataset ratings are on a 1 to 10 scale with 10 expressing the highest preference. The preferences were re-scaled into the [1 - 5] scale to be suitable for the proposed framework. The ratings 10 and 9 are converted to 5, and the ratings 8 and 7 to 4, and the ratings 6 and 5 to 3, and the ratings 4 and 3 to 2, and the ratings 2 and 1 to 1.

Experiment parameters

1. **Threshold-based value:** The Pearson value for selecting the candidate neighbours varies from 0 to 1.
2. **Top-N value:** The top-N strategy is based on the degree of overlaps between users – number of shared rated items. The similar neighbours are ranked according to their degree of overlap, then the most 10 users are selected.
3. **Number of clusters:** The number of clusters has a significant impact on the prediction quality. The experiments were performed with a different number of clusters k where k varies from 5 to 50 clusters.

4.4 Experimental Results

The Pearson nearest neighbour algorithm is used for prediction. Table 4.2 presents the Mean Absolute Error (MAE) for the target users.

User	**7346**	**8067**	**31556**	**60244**	**60707**	**75591**	**76499**	**97874**	**100459**	**100906**	**AVG**
MAE	0.55208	0.8	0.83333	0.67879	0.60417	0.73214	1.03788	1.375	0.39167	0.61538	**0.76204**

Table 4.2: MAE for the K Nearest Neighbour filtering method

Ten experiments on the dataset with a different number of clusters were conducted using explicit feedback data. Table 4.3 presents the Mean Absolute Error (MAE) for the target users considering the different number of clusters k where k ∊ {5, 10, 15, 20, 25, 30, 35, 40, 45, 50. Also, the experiments were repeated but instead of using only explicit feedback data, implicit feedback data is added to the clusters (submatrices). Table 4.4 presents the Mean Absolute Error (MAE) for the target users after adding the implicit feedback data.

Method T.User	k= 5	k= 10	k= 15	k= 20	k= 25	k= 30	k= 35	k= 40	k= 45	k= 50
7346	0.73077	0.70833	0.72727	0.72324	0.73333	0.71429	0.77778	0.66667	0.77778	0.75
8067	1.10512	0.75	0.78	1.02	0.45714	0.54762	0.55556	0.54762	0.65473	0.71429
31556	1.16667	1.5	1.5	1	1.02451	0.66667	1.00000	1.12500	1.83333	1.16667
60244	0.60753	0.56349	0.55357	0.816	0.69815	0.63043	0.53333	0.63750	0.64035	0.84375
60707	0.59352	0.67879	0.82554	0.89164	0.95872	0.92775	0.85901	0.76921	1.02771	0.72519
75591	0.74251	0.81061	0.66667	0.64517	0.5	0.67541	0.56742	0.76812	0.66667	0.85412
76499	0.28952	0.12874	0.12012	0.42158	0.25473	0.32451	0.49856	0.21215	0.00000	0.42513
97874	0.86364	1.21548	0.71429	0.71429	0.65897	0.21415	0.69584	0.58941	0.21498	0.21458
100459	0.6	0.47917	0.65556	0.455	0.79821	0.68954	0.66667	0.78456	0.56847	0.66667
100906	0.75	0.33333	0.6	0.75	0.96296	1	0.47143	0.46190	0.27273	0.33333
AVG	**0.74493**	**0.71679**	**0.71430**	**0.74369**	**0.70467**	**0.63904**	**0.66256**	**0.65621**	**0.66567**	**0.66937**

Table 4.3: MAE for clustering explicit feedback data (k = 5 to 50)

Method / T.user	k= 5	k= 10	k= 15	k= 20	k= 25	k= 30	k= 35	k= 40	k= 45	k= 50
7346	1	0.53333	0.46429	0.5	0.575	0.7	0.48333	0.49856	0.5	0.52564
8067	1.25	0.5	0.62054	0.80556	0.6	0.125	0.68273	0.35714	0.35210	0.57407
31556	1.16667	1	1	0.5883	0.33333	0.725	0.75655	0.78294	0.65912	0.66667
60244	0.55556	0.57464	0.58824	0.5	0.53186	0.60294	0.51222	0.54896	0.59472	0.49825
60707	0.57847	0.8324	0.83945	0.87795	0.96022	0.89573	0.84129	0.58333	0.45512	0.32515
75591	0.74358	0.79464	0.75	0.69697	0.59375	0	0.40258	0.49856	0.49877	0.46891
76499	0.56522	0.67647	0.46154	0.52778	0.36842	0.33333	0.38871	0.43333	0.53846	0.57692
97874	0.86667	1.42857	0.63704	0.71429	0.75	1	0.65891	0.65874	0.58333	0.5
100459	0.38095	0.38922	0.33333	0.38451	0.6859	1	0.43222	0.58411	0.57143	0.49825
100906	0.41667	0.33333	0.4	0.5	0.65569	0.57143	0.55714	0.29246	0.46667	0.49861
AVG	**0.75238**	**0.70626**	**0.60944**	**0.60953**	**0.60542**	**0.59534**	**0.57157**	**0.52381**	**0.52197**	**0.51325**

Table 4.4: The MAE for clustering feedback data (k = 5 to 50)

(After adding implicit feedback data)

Tables 4.5 (a), 4.5 (b), 4.5 (c), 4.5 (d), 4.5 (e),4.5 (f), 4.5 (g), 4.5 (h), 4.5 (i) and 4.5 (j) show the sub-matrices sparsity level after clustering users. The clusters mentioned in the tables are the clusters that produced after applying the clustering process. For reviewing the distribution of users among the clusters, see (Appendix D).

Cluster #	1	3	4
Sparsity level	99.39%	99.91%	98.98%

(a): Sparsity level after clustering (No. of clusters = 5)

Cluster #	0	1	5	7	9
Sparsity level	98.42%	99.91%	99.04%	97.31%	99.49%

(b): Sparsity level after clustering (No. of clusters = 10)

Cluster #	2	3	6	10	12	13
Sparsity level	99.19%	99.91%	96.93%	99.24%	98.78%	98.3%

(c): Sparsity level after clustering (No. of clusters = 15)

Cluster #	1	3	4	11	16
Sparsity level	98.16%	99.21%	96.95%	99.36%	99.91%

(d): Sparsity level after clustering (No. of clusters = 20)

Cluster #	3	4	8	10	15	20	24
Sparsity level	99.91%	98.38%	99.15%	98.11%	97.76%	97.72%	97.86%

(e): Sparsity level after clustering (No. of clusters = 25)

Cluster #	4	9	12	15	23	27	29
Sparsity level	98.12%	84.86%	96.8%	99.91%	99.48%	95.8%	99.19%

(f): Sparsity level after clustering (No. of clusters = 30)

Cluster #	9	11	17	22	24	27
Sparsity level	98.55%	92.24%	99.16%	98.16%	99.91%	98.65%

(g): Sparsity level after clustering (No. of clusters = 35)

Cluster #	1	7	12	15	26	32	37
Sparsity level	99.08%	99.22%	97.14%	98.86%	99.91%	97.89%	96.26%

(h): Sparsity level after clustering (No. of clusters = 40)

Cluster #	7	11	21	23	25	26	37
Sparsity level	97.99%	98.79%	96.6%	99.53%	99.91%	82.52%	91.81%

(i): Sparsity level after clustering (No. of clusters = 45)

Cluster #	3	8	11	12	15
Sparsity level	99.91%	94.22%	98.14%	96.59%	99.23%

(j): Sparsity level after clustering (No. of clusters = 50)

Tables 4.5. Sparsity level after clustering explicit feedback data

Tables 4.6 (a), 4.6 (b), 4.6 (c), 4.6 (d), 4.6 (e), 4.6 (f), 4.6 (g), 4.6 (h), 4.6 (i) and 4.6 (j) show the number of implicit rating added to the clusters and the sparsity level before and after adding the implicit rating.

Cluster #	1		3		4	
Number of implicit ratings	55089		101211		8692	
Sparsity level	99.39%	98.43%	99.91%	99.76%	98.98%	98.18%

(a): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 5)

Cluster #	0		1		5		7		9	
Number of implicit ratings	4520		101211		6856		79467		25132	
Sparsity level	98.42%	97.55%	99.91%	99.78%	99.04%	98.24%	97.31%	95.1%	99.49%	98.76%

(b): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 10)

Cluster #	2		3		6		10		12		13	
Number of implicit ratings	11280		76891		2387		11762		4097		2172	
Sparsity level	99.19%	98.2%	99.91%	99.78%	96.93%	95.12%	99.24%	98.34%	98.78%	97.88%	98.3%	97.52%

(c): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 15)

Cluster #	1		3		4		11		16	
Number of implicit ratings	1497		12090		2755		8604		70073	
Sparsity level	98.16%	97.44%	99.21%	98.23%	96.95%	95.11%	99.36%	98.76%	99.91%	99.78%

(d): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 20)

Cluster #	3		4		8		10		15		20		24	
Number of implicit ratings	82102		1725		4880		1324		2380		2508		2056	
Sparsity level	99.91%	99.77%	98.38%	97.51%	99.15%	98.3%	98.11%	97.35%	97.76%	96.21%	97.72%	96.21%	97.86%	96.68%

(e): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 25)

Cluster #	4		9		12		15		23		27		29	
Number of implicit ratings	1676		156		1353		62589		13883		604		5262	
Sparsity level	98.12%	97.22%	84.86%	83.08%	96.8%	95.41%	99.91%	99.8%	99.48%	98.88%	95.8%	94.58%	99.19%	98.39%

(f): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 30)

Cluster #	9		11		17		22		24		27	
Number of implicit ratings	2133		568		4275		1231		72833		2358	
Sparsity level	98.55%	97.67%	92.24%	89.64%	99.16%	98.46%	98.16%	97.52%	99.91%	99.78%	98.65%	97.69%

(g): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 35)

Cluster #	1		7		12		15		26		32		37	
Number of implicit ratings	5431		6194		605		2544		70785		849		536	
Sparsity level	99.08%	98.26%	99.22%	98.42%	97.14%	96.15%	98.86%	98.13%	99.91%	99.78%	97.89%	97.17%	96.26%	95.25%

(h): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 40)

Cluster #	7		11		21		23		25		26		37	
Number of implicit ratings	759		2426		236		12767		63800		200		677	
Sparsity level	97.99%	97.38%	98.79%	98.04%	96.6%	96.06%	99.53%	98.98%	99.91%	99.79%	82.52%	78.81%	91.81%	88.96%

(i): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 45)

Cluster #	3		8		11		12		15	
Number of implicit ratings	73496		408		840		525		6720	
Sparsity level	99.91%	99.78%	94.22%	92.24%	98.14%	97.5%	96.59%	95.66%	99.23%	98.37%

(j): Number of implicit ratings added to clusters & sparsity level before and after clustering (No. of clusters = 50)

Tables 4.6: Number of implicit ratings added to clusters,

& sparsity level before and after adding implicit ratings

4.5 Results discussion

On analysing the proposed framework performance, in the subsection 4.5.1, the proposed framework prediction results are compared with the baseline approach (K NN) results. In the subsection 4.5.2, the relation between clustering and sparsity level will be discussed.

4.5.1 Impact of the Proposed Framework on the Accuracy of Recommendations

The proposed framework had been tested with adifferent number of clusters using implicit feedback data and without using implicit feedback data. It can be observed from figure 4.1 and 4.2, that each MAE value for the experiments without using implicit feedback and by using implicit feedback was lower than the K NN filtering method. These values reveal that using clustering explicit feedback data with and without adding implicit feedback data achieve good quality compared with K Nearest Neighbours filtering method.

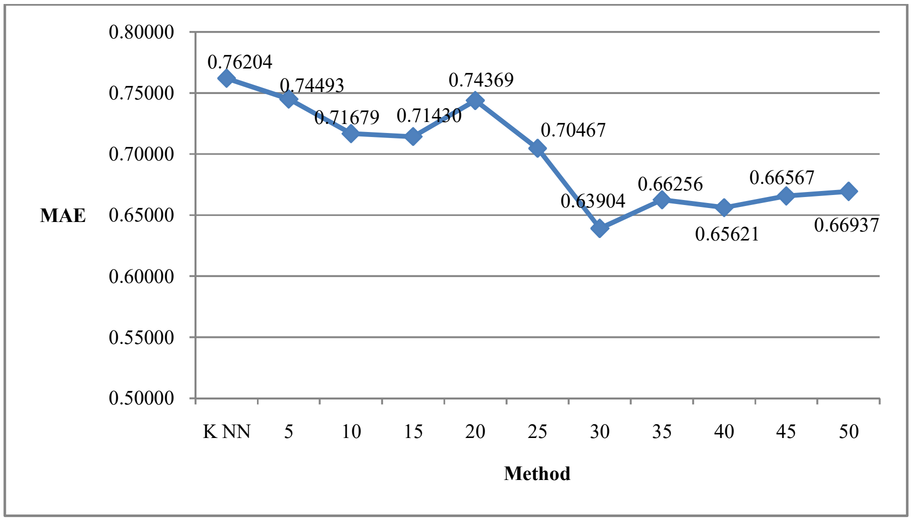


Fig 4.1: MAE for clustering explicit feedback data compared to K NN method

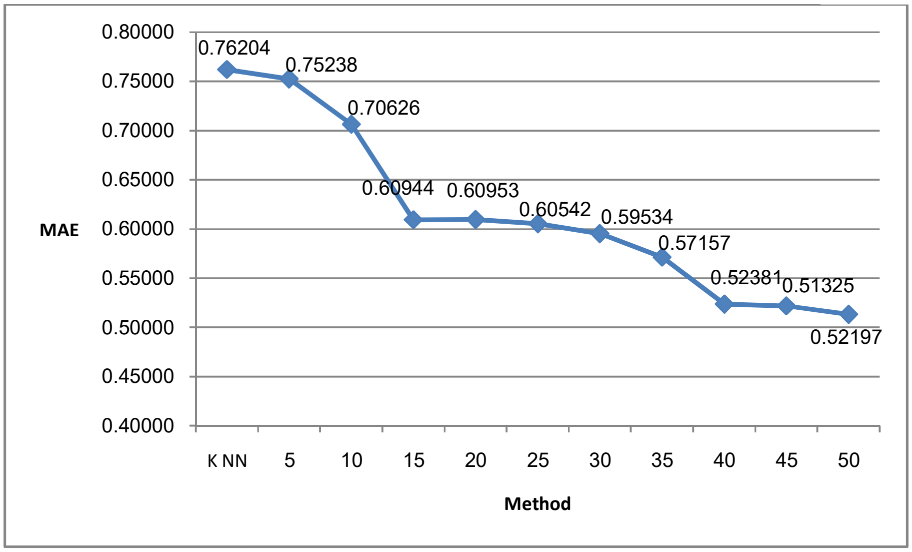


Fig 4.2: MAE for clustering explicit & implicit feedback data compared to K NN method

Figure 4.3 shows the ten experiments results, it can be observed from the figure that most the experiments "clustering explicit and implicit feedback data" (9 out of 10 experiments) achieved better quality compared to "clustering only explicit feedback data".

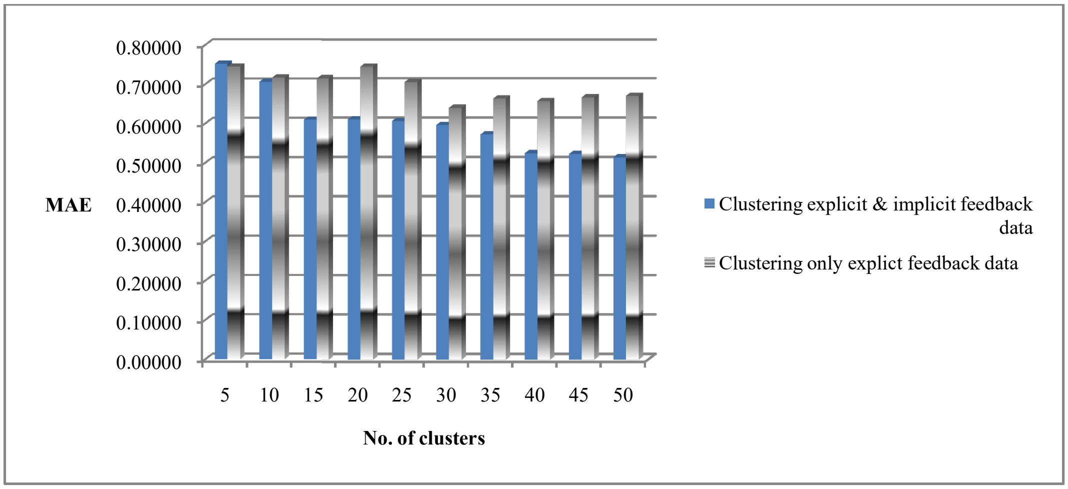


Fig 4.3: MAE for clustering explicit feedback data compared to clustering explicit & implicit feedback data

Table 4.7 concludes all the MAE average resulted from the experiments: 1) clustering explicit feedback data, 2) clustering explicit and implicit feedback data, and 3) K Nearest Neighbours-method. It can be observed that clustering explicit feedback performs better than the K Nearest Neighbours method based on the Pearson correlation algorithm (0.69172, 0.76204 respectively). Also, it can be observed that taking into account implicit feedback data can contribute in the prediction recommendation and result in more accurate prediction than clustering only explicit feedback data and Nearest Neighbours filtering method (0.6009, 0.69172, 0.76204 respectively). Clustering explicit feedback data achieved up to 9.22% improvement over the K NN method. The proposed framework (clustering explicit & implicit feedback data) performs as much as 13.12% better than clustering only explicit feedback data, and 21.14% better than K Nearest Neighbours filtering method. The prediction accuracy for the proposed framework, and clustering explicit feedback data, and K Nearest Neighbours filtering method are given in figure 4.4. Clearly, we can conclude that the proposed framework gives better prediction accuracy than the accuracy of clustering only explicit feedback data and the accuracy of the K Nearest Neighbours filtering method.

Method	MAE
K NN method	0.76204
Clustering-only	0.69172
Proposed framework	0.6009

Table 4.7: MAE results for the three methods

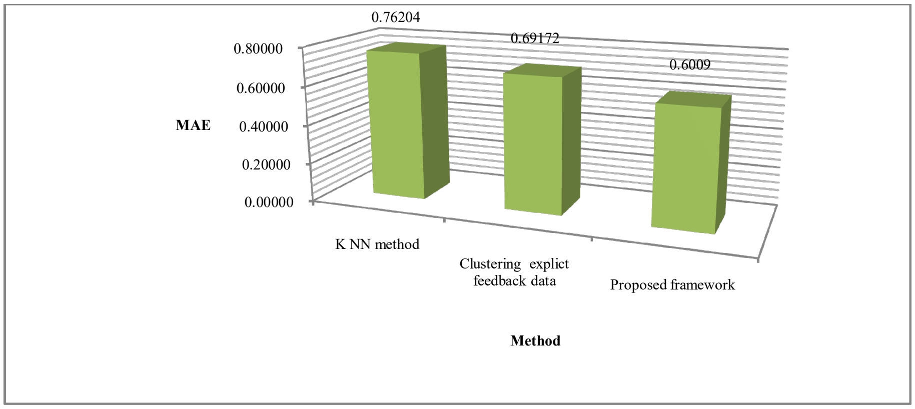


Fig. 4.4: Comparison of recommendation accuracy improvement

As shown in Table 4.8 each MAE value of the ten experiments (k = 5 to 50) were lower than the MAE for the K NN filtering method (0.76204).

No. of clusters	k= 5	k= 10	k= 15	k= 20	k= 25	k= 30	k= 35	k=40	k= 45	k= 50	**AVG**
MAE	0.75238	0.70626	0.60944	0.60953	0.60542	0.59534	0.57157	0.52381	0.52197	0.51325	
Improvement over baseline %	1.27	7.32	20.03	20.01	20.55	21.88	24.99	31.26	31.5	32..64	**21.14**

Table 4.8: Improvement over baseline for the proposed framework (k= 5 to 50)

4.5.2 Impact of proposed framework on sparsity level

The sparsity degree of evaluation in the matrix has an important impact on the performance of collaborative filtering systems. We can notice the impact of the proposed framework on the sparsity level of the sub-matrices by measuring the sparsity level after applying the framework method. Tables 4.5 presents the sparsity level after applying the framework processes – except adding implicit feedback data - on the original matrix (original matrix sparsity level= 99.86%). Tables 4.6 presents the sparsity level after after adding implicit feedback data to the clusters (sub-matrices). As can be seen from tables 4.5 that the sparsity level was decreased for the majority of the sub-matrices. Gray cells in Tables 4.5 represent the sub-matrices that the sparsity level increased in it, which is about 17.24% of the total sub-matrices and the increase of the sparsity level in worst case was not more than 0.05%. From Tables 4.6 it can be observed that the sparsity level is decreased in all sub-matrices after adding implicit feedback data. The decreased is between (0.07% and 26.71%) in the sub-matrices. Clearly, we can conclude that using the proposed framework produce sub-matrices much denser than the original matrix. Thus, it is highly useful to use the combination of explicit and implicit feedback data with clustering data mining technique for the recommendation process. Figures 4.5(a), 4.5(b), 4.5(c), 4.5(d), 4.5(e) , 4.5(f) , 4.5(g) , 4.5(h) , 4.5(i), and 4.5(j) show the impact of the sparsity level on the quality of recommendations. It can be observed that the prediction accuracy for users is better in clusters with less sparsity level, while it is less in clusters with high sparsity level.

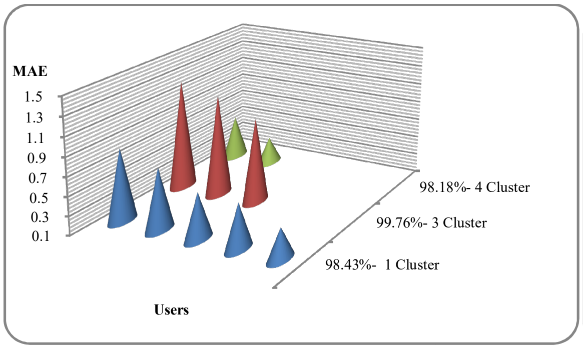


Fig. 4.5(a): No. of clusters = 5

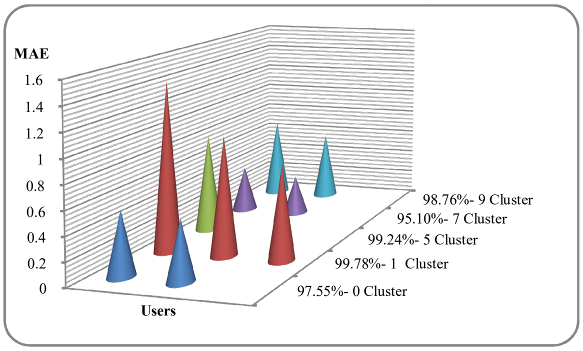


Fig. 4.5 (b): No. of clusters = 10

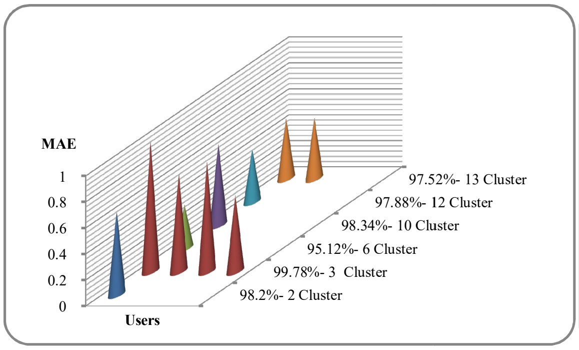


Fig. 4.5 (c): No. of clusters = 15

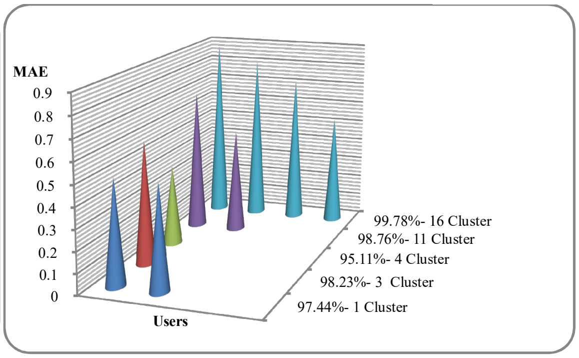


Fig. 4.5 (d): No. of clusters = 20

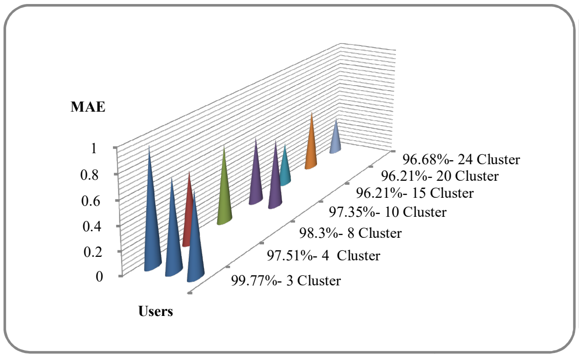


Fig. 4.5 (e): No. of clusters = 25

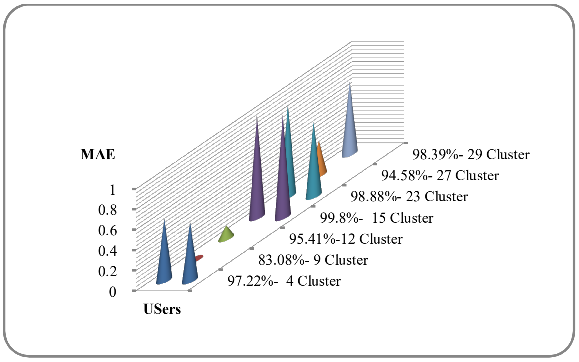


Fig. 4.5 (f): No. of clusters = 30

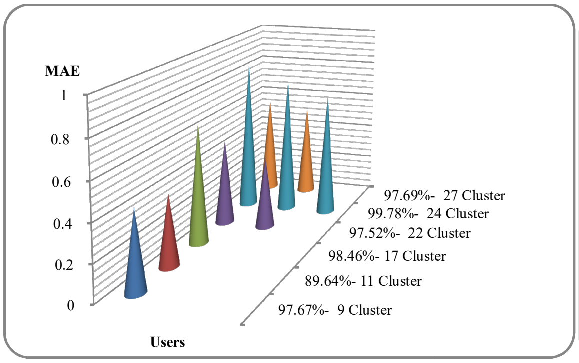


Fig. 4.5(g): No. of clusters = 35

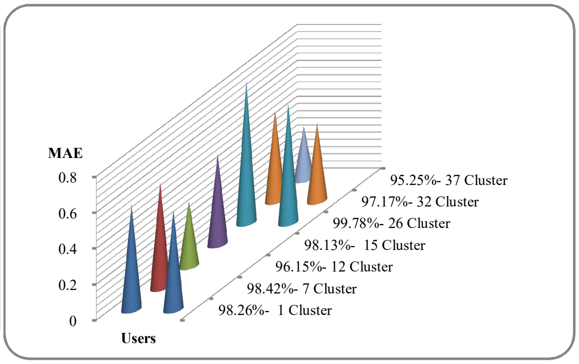


Fig. 4.5 (h): No. of clusters = 40

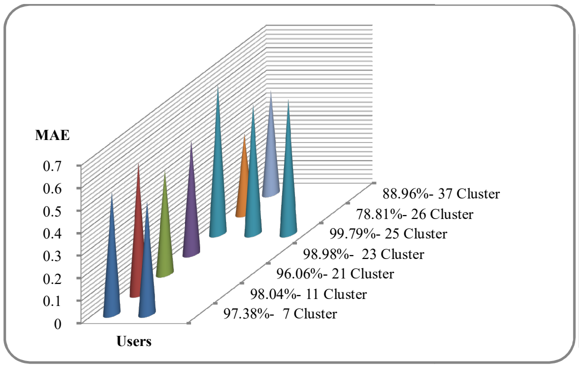


Fig. 4.5 (i): No. of clusters =45

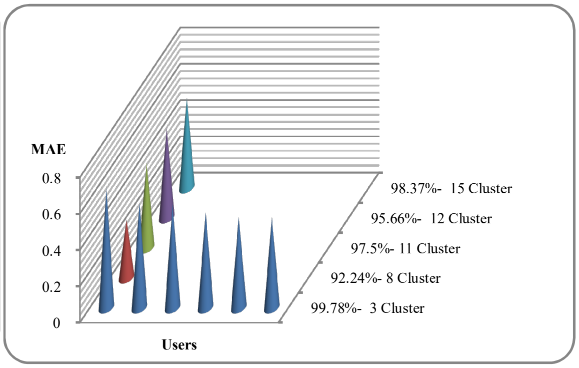


Fig. 4.5 (j) : No. of clusters = 50

4.6 Summary

This chapter assessed whether the proposed framework leads to more accurate recommendation results compared to the memory-based collaborative method. The evaluation employed a realdataset (book-crossing dataset) for testing the performance of the proposed framework. The Mean Absolute Error (MAE) is used to measure the quality of the proposed framework. Two tasks are achieved for evaluating the proposed framework. In the first task, the quality of recommendations generated using clustering explicit feedback data is compared to recommendations generated by the standard K Nearest Neighbours (KNN) method. In the second task, the quality of recommendations generated using the proposed framework is compared to the recommendations generated by clustering clustering only explicit feedback data, and compared to recommendations generated by the standard K Nearest Neighbours (KNN) method. The comparison implies that the proposed framework result in more accurate prediction than clustering only explicit feedback data, and the standard K Nearest Neighbours (KNN) method.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

The rapid expansion of Web users has raised the importance of the Web-based applications. Academic D-libraries use Web as an interactive medium to allow users access its contents. With the development of Web-based academic D-libraries resources provided by Web is growing rapidly. This huge number of resources provide opportunities to students, at the same time, students have to face the problem of the overload information which affects the efficiency of information seeking and leads to waste of time in the usage of the digital library.

D-library systems usually use collaborative filtering recommender systems for easy access to relevant information resources. Collaborative filtering relies on obtaining feedback data to provide recommendations. The feedback can either be explicit feedback, e.g. ratings, or implicit feedback, e.g. clicks. It is well known that the success of collaborative filtering recommendation systems mainly depends on sufficient feedback data support. Usually students do not put extra amount of effort to provide explicit feedback which results in a sparsity evaluations. Due to the data sparseness probable, the similarity between students cannot be defined, even when the similarity is defined the similarity computation is imprecise.

The main research was focused on recommending relevant resources in D-library. The study objectives were:

- Highlight the importance of the Web knowledge in academic D-library.
- Using the usage knowledge as a collaborative information source.
- Emphasize the use of clustering feedback data for collaborative filtering.
- Capitalizing on the efficiency of user behaviour.

To achieve these objectives a Web usage mining framework based on collaborative filtering approach is proposed. The proposed framework consists of two components, Off-line and On-line components. The first component composed of two stages: data pre-processing and the derivation of student clusters. In the first stage implicit data are turned into numeric ratings, then this data is added to the user–resource matrix. In the second stage, the k-means clustering algorithm is applied to the user–resource matrix to group students. The online component is composed of two stages: The first stage is responsible for constructing the active student profile. The second stage is responsible for constructing the recommendation list, this stage consists of three phases, in the first phase the cosine similarity function was used to classify the active student to the best cluster. In the second phase the Pearson correlation coefficient function was used to measure the similarity between the active student and students in the selected cluster, then, the students who share the highest similarity with the current student are chosen to be used as a source for generating the recommendations. Finally, a recommendations list is provided.

This proposed framework offer clustering technique incorporated with implicit feedback data for performing the density of the student-resources matrix used for generating recommendations. Invalidating the hypotheses that it is using the implicit feedback data and the explicit feedback data as a collaborative information source with clustering technique will yield more useful recommendations. Two measures are used to validate the hypotheses:

- Mean Absolute Error (MAE) metric: This metric intends to measure the quality of recommendations.
- Sparsity level metric: This metric intends to measure the Sparsity level of the matrix. The sparsity degree of evaluation in the matrix has an important impact on the performance of the collaborative filtering systems. As the level of data sparsity decreases, the recommendation accuracy gets better.

The first hypothesis we sustain that it is "H1: Using Web clustering mining technology significantly produces effective recommendation".

The proposed framework utilizes clustering data mining approach to alleviate the sparsity problem. The clustering technique is used to divide the sparse student-resource matrix into sub-matrices much denser than the original matrix and therefore can be used much more efficiently than the original sparse matrix. Furthermore, generating recommendations based on clusters produce better results because students belong to the same cluster usually have similar interests.

To validate the hypothesis an experimental validation includes: conducting ten experiments with a different number of clusters k (k = {5, 10, 15, 20, 25, 30, 35, 40, 45, 50}). The clustering technique is applied on the explicit feedback data. Then, the quality of the recommendations generated by these experiments is compared with recommendations generated by the standard K Nearest Neighbours method (KNN). The Mean Absolute Error (MAE) metric is examined for the experiments. The experimental results indicated that the mean absolute error (MAE) is significantly better when using k-means clustering technique compared to the K Nearest Neighbours method, whereas the MAE is 0.69172 using k-means, and 0.76204 for the K Nearest Neighbours method (K NN). The clustering method achieved up to 9.22% improvement over the K NN method. These results indicate that using one cluster (original matrix) that include a very high number of neighbours, they are less similar degrades prediction quality compared with using sub-matrices that include fewer neighbours, which are highly similar.

The Sparsity level equation is used to measure the sparsity of the original matrix and the resulted matrices after applying clustering approach on the original matrix. The feature dimensionality reduction achieved by applying the clustering technique. The sparsity level results show that the sparsity level is reduced on (82.76%) of the sub-matrices.

The second hypothesis we sustain that it is "H2: Recommendation approach that considers the implicit feedback data will result in more accurate recommendations".

The principal motivation for using implicit feedback is that it: a) provides the system with large quantities of feedback data rather than the sparse data encountered by explicit user feedback, b) is immediately available, and c) can achieve much greater coverage for resources than was achieved by using explicit data. The study identified five actions include "printing", "bookmarking", "downloading", "reading" and "viewing abstract" each action can be used as an indicator of the student preferences. These actions had given an appropriate numerical value representing the importance of it, where a higher value means higher evidence of interest. The numerical value ratings are between "1 - 5". The values "1 to 4" treated a spositive feedback, and the value "1"treated asnegative feedback.

To validate the hypothesis the Mean Absolute Error (MAE) metric is examined throughout using only explicit feedback data compared to using the proposed framework (a combination of explicit and implicit feedback data.). Ten experiments were conducted with a different number of clusters k (k = {5, 10, 15, 20, 25, 30, 35, 40, 45, 50}). First, the clustering applied on explicit feedback data, then the experiments are repeated by adding implicit feedback data to the sub-matrices. The experimental results showed that using the implicit feedback data obtained better results than using only explicit feedback data, whereas the MAE is 0.69172 by using explicit feedback data, and 0.6009 by using explicit feedback data with implicit feedback data. By using explicit and implicit feedback data the performance is as much as 13.12% better than clustering only explicit feedback data, and 21.14% better than K Nearest Neighbours method.

The sparsity level equation is used to measure the sparsity of the original matrix and the resulted matrices after using the proposed framework. The sparsity level results show that the sparsity level is reduced between (0.07% and 26.71%), whereas the sparsity level is between 99.79% and 78.81% in the sub-matrices when using framework, and 99.86% (original matrix) before applying the proposed framework.

The overall results show that the proposed framework can alleviate the Sparsity problem resulting in improving the accuracy of the recommendations. The reason is that, the recommendations are produced from clusters that include fewer neighbours, they are highly similar with sufficient feedback data which improve the prediction quality, unlike using one cluster (original dataset) that include a very high number of neighbours, they are less similar with insufficient feedback data which degrades prediction quality.

5.2 Future Work

Collaborative filtering recommender systems are playing a major role in the Web academic D-libraries revolution. They are helping library users efficiently manage content overload; these systems suffer from sparsity problem. Therefore, there is a need to develop methods to solve the problem of sparsity in the collaborative filtering for academic D-libraries.

Sparsity problem in D-library recommender systems is still a very active research area and, new advanced approaches will appear in future.

In this work a collaborative filtering system based on the feedback data is built, although this method alleviates sparsity problem, using such method can be problematic when little explicit and implicit feedback data is available. There exist two data sources in academic D-libraries, the first source, library management database, the second source, Web server files. Demographics student’s information from library management database and explicit and implicit feedback data from Web server files can be used together in a recommender system. This results in a hybrid system that uses students' demographic information data and students' feedback data. Integrating Collaborative Filtering (CF) with Demographic recommender (DM) recommendations will alleviate the sparsity problem and could provide more reasonable recommendation results. The general process of the Hybrid system (CF&DM)is shown in figure 5.1.

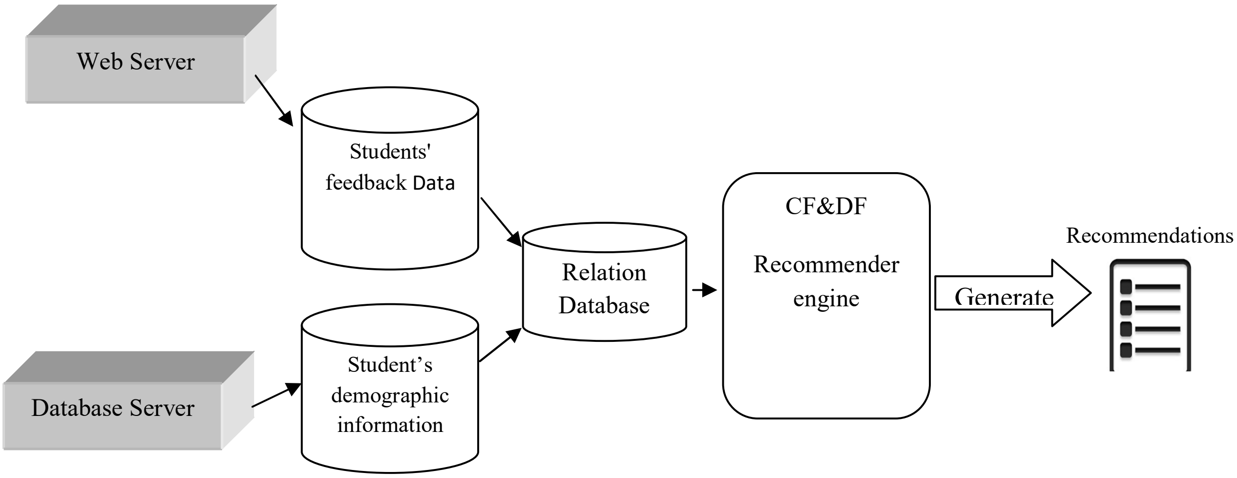


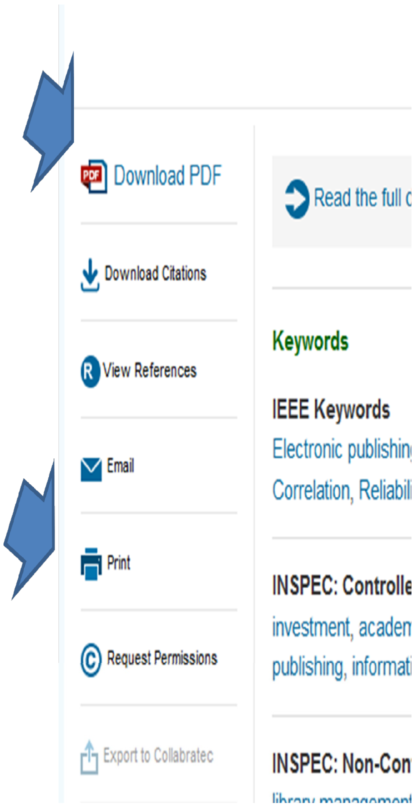
Fig. 5.1: Integration of demographic data and feedback data

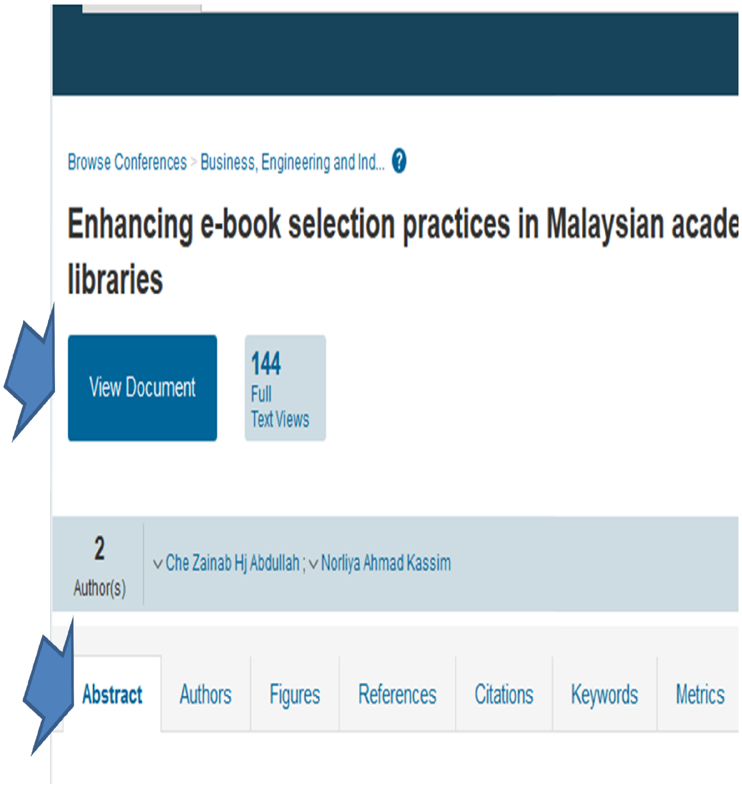
Incorporating student’s demographic information and feedback data into a framework will be a new way for academic D-library to recommend relevant information. The students demographic involve information about the students, e.g. who online students are, where they study, what grade are they in? In the context of collaboration approach, these demographics information can be used to discover useful knowledge such as "The students of certain specialty are likely to prefer some kinds of resources preferred by similar students"," The students of certain specialty and grade are likely to prefer some kinds of resources preferred by similar students”. Also using demographic information could lead to solving the Cold startproblem. For example, students registering Database course would all need the same materials; if first students needs and behaviour observed, then later students in the same group can be directed to useful resources easily without identifying their preferences explicitly.

Appendix A

User behaviours

A.1 IEEE Digital library:

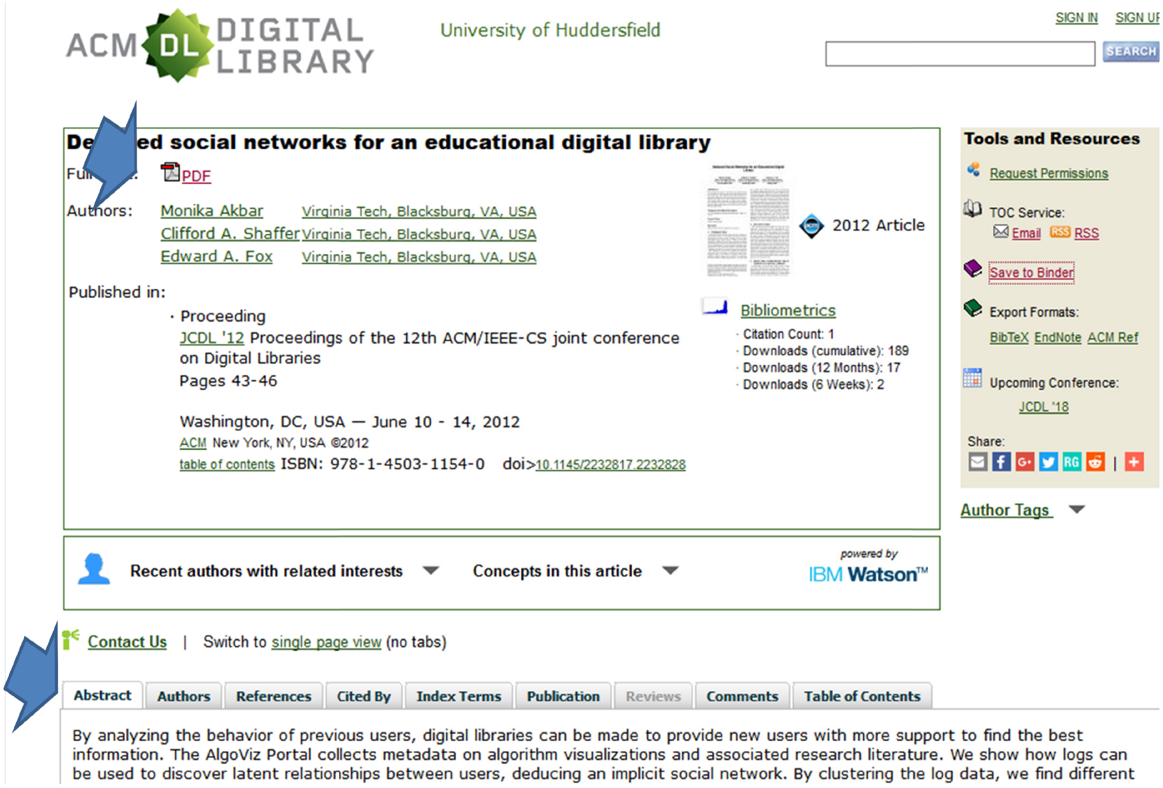




Enhancing e-book selection practices in Malaysian acade libraries

Che Zainab Hj Abdullah ; Norliya Ahmad Kassim

A.2 ACM Digital library:



De ed social networks for an educational digital library

Ful : PDF

Authors: Monika Akbar Virginia Tech, Blacksburg, VA, USA
Clifford A. Shaffer Virginia Tech, Blacksburg, VA, USA
Edward A. Fox Virginia Tech, Blacksburg, VA, USA

2012 Article

Published in:
· Proceeding
JCDL '12 Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries
Pages 43-46

Washington, DC, USA — June 10 - 14, 2012
ACM New York, NY, USA ©2012
table of contents ISBN: 978-1-4503-1154-0 doi>10.1145/2232817.2232828

Bibliometrics
· Citation Count: 1
· Downloads (cumulative): 189
· Downloads (12 Months): 17
· Downloads (6 Weeks): 2

Tools and Resources
Request Permissions
TOC Service:
Email RSS
Save to Binder
Export Formats:
BibTeX EndNote ACM Ref
Upcoming Conference:
JCDL '18
Share:

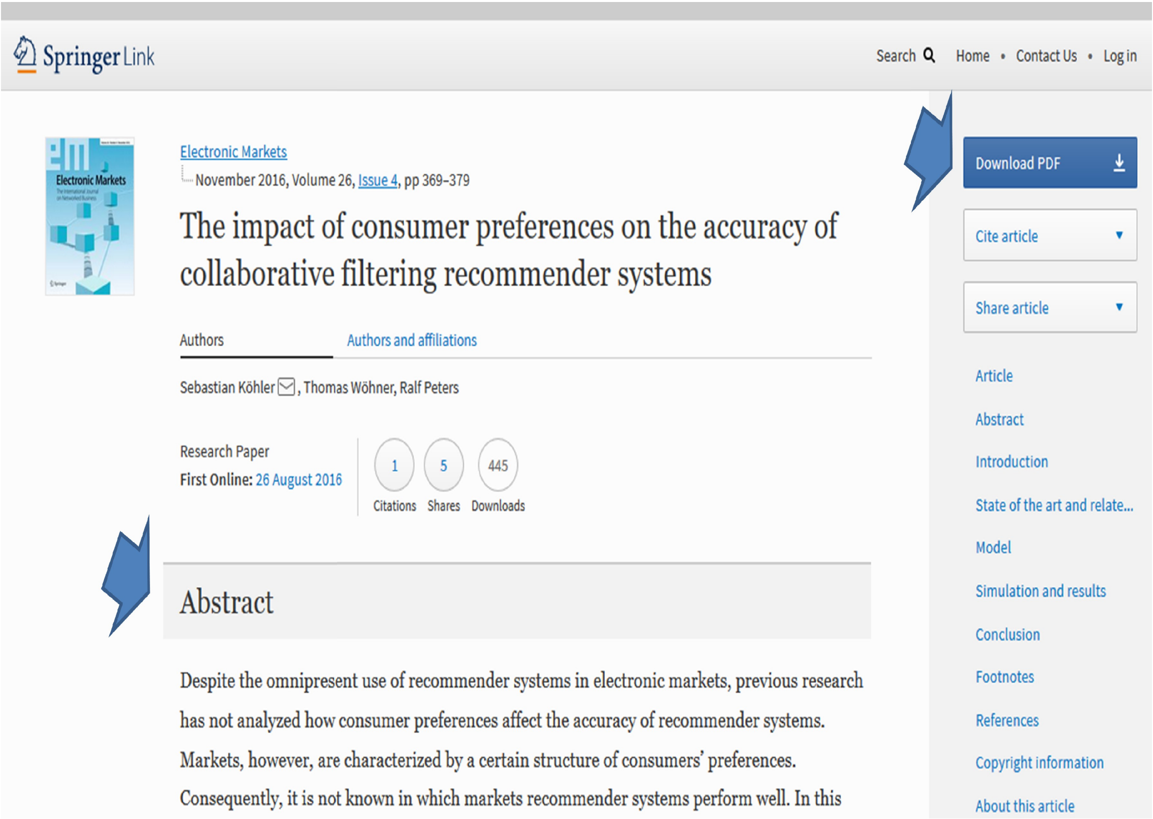
Author Tags

Recent authors with related interests   Concepts in this article   powered by IBM Watson™

Contact Us | Switch to single page view (no tabs)

By analyzing the behavior of previous users, digital libraries can be made to provide new users with more support to find the best information. The AlgoViz Portal collects metadata on algorithm visualizations and associated research literature. We show how logs can be used to discover latent relationships between users, deducing an implicit social network. By clustering the log data, we find different

A.3 Springer Digital library:



Electronic Markets

November 2016, Volume 26, Issue 4, pp 369–379

The impact of consumer preferences on the accuracy of collaborative filtering recommender systems

Authors | Authors and affiliations

Sebastian Köhler, Thomas Wöhner, Ralf Peters

Research Paper

First Online: 26 August 2016

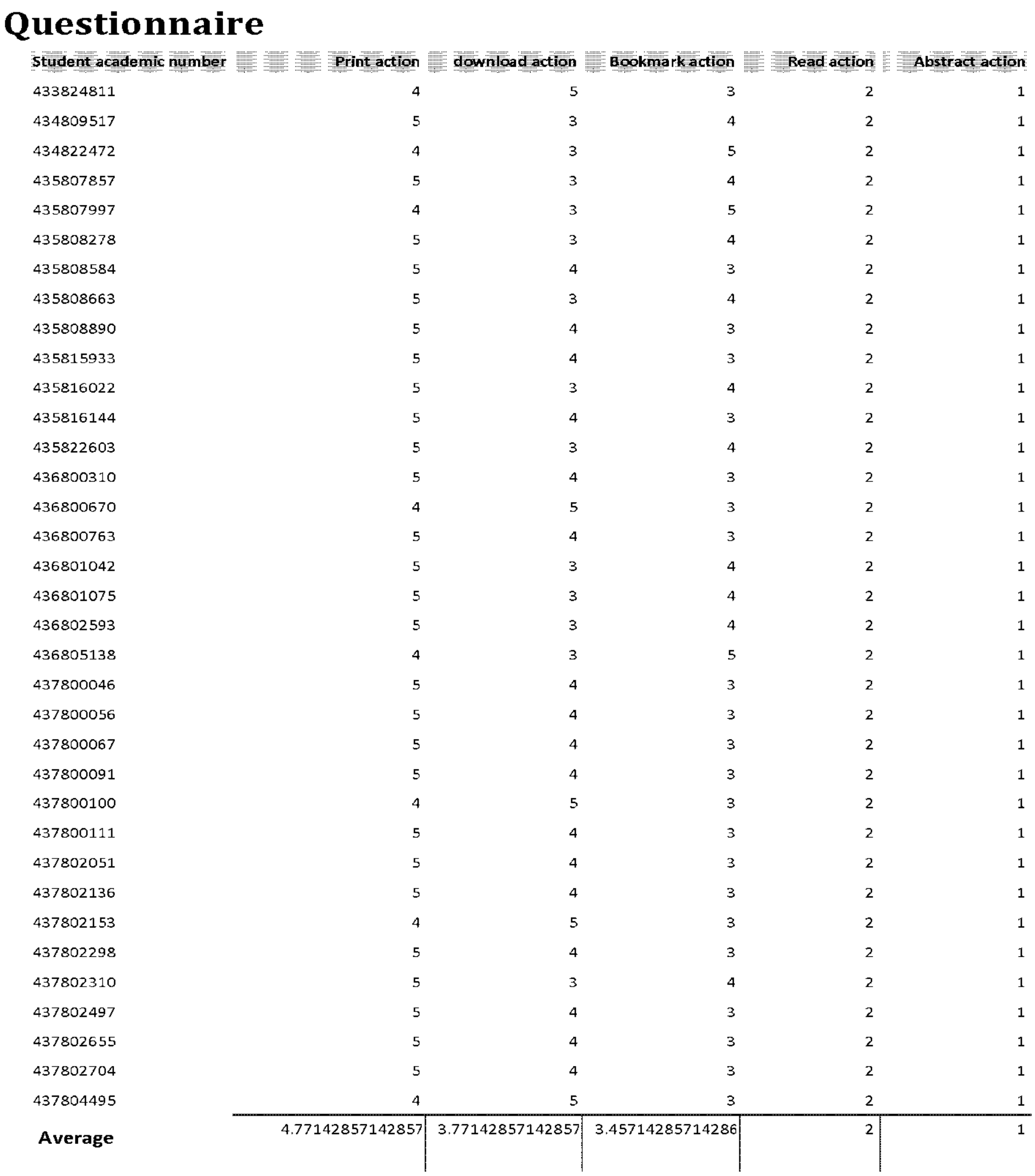
1	5	445
Citations	Shares	Downloads

Abstract

Despite the omnipresent use of recommender systems in electronic markets, previous research has not analyzed how consumer preferences affect the accuracy of recommender systems. Markets, however, are characterized by a certain structure of consumers' preferences. Consequently, it is not known in which markets recommender systems perform well. In this

Appendix B

Students’ Questionnaire



Questionnaire

Student academic number	Print action	download action	Bookmark action	Read action	Abstract action
433824811	4	5	3	2	1
434809517	5	3	4	2	1
434822472	4	3	5	2	1
435807857	5	3	4	2	1
435807997	4	3	5	2	1
435808278	5	3	4	2	1
435808584	5	4	3	2	1
435808663	5	3	4	2	1
435808890	5	4	3	2	1
435815933	5	4	3	2	1
435816022	5	3	4	2	1
435816144	5	4	3	2	1
435822603	5	3	4	2	1
436800310	5	4	3	2	1
436800670	4	5	3	2	1
436800763	5	4	3	2	1
436801042	5	3	4	2	1
436801075	5	3	4	2	1
436802593	5	3	4	2	1
436805138	4	3	5	2	1
437800046	5	4	3	2	1
437800056	5	4	3	2	1
437800067	5	4	3	2	1
437800091	5	4	3	2	1
437800100	4	5	3	2	1
437800111	5	4	3	2	1
437802051	5	4	3	2	1
437802136	5	4	3	2	1
437802153	4	5	3	2	1
437802298	5	4	3	2	1
437802310	5	3	4	2	1
437802497	5	4	3	2	1
437802655	5	4	3	2	1
437802704	5	4	3	2	1
437804495	4	5	3	2	1
Average	4.77142857142857	3.77142857142857	3.45714285714286	2	1

Appendix C

Obtaining Numerical Values for the Implicit Rating

“Source Code”

```
// ############################ Cluster opinion // ############################
double sumop = 0;
 int T = 0;
20
int qqq;
for (int w = 2; w < isbncnt + 2; w++)
{
for (qqq = 0; qqq < usercnt; qqq++)
{
if (usersinclusters[qqq, 0] == clname)
{
if (usersinclusters[qqq, w] != 0)
{
sumop = sumop + usersinclusters[qqq, w];
T = T + 1;
}
}
}
clusterOpinion[clname, w] = sumop / T;
sumop = 0;
T = 0;
}
// ############################ Cluster opinion // ############################

/ ################# Calculating Sparsity for the chosen cluster before adding implicit Data  ###############//
double nonZB = 0;
Totalcells = 0;
Sparsitycluster = 0;
SparsityclusterImplicity = 0;
int cntisbn=0;
for (int w = 2; w < isbncnt + 2; w++)
{
  for ( int q = 0; q < usercnt; q++)
  {
    if (usersinclusters[q, 0] == clname)
  {
  if (usersinclusters[q, w] != 0)
 {
Totalcells = Totalcells + 1;
isbnclusterarray[cntisbn, 0] = isbnarray[w - 2];
isbnclusterarray[cntisbn, 1] = Convert.ToString(w - 2);
isbnclusterarray[cntisbn, 2] = Convert.ToString(clusterOpinion[clname, w]);
cntisbn++;
break;
}
}
}
}
for (int q = 0; q < usercnt; q++)
{
   if (usersinclusters[q, 0] == clname)
   {
     for (int w = 2; w < isbncnt + 2; w++)
    {
     if (usersinclusters[q, w] != 0)
         nonZB = nonZB + 1;
     }
}
}
Sparsitycluster = 1 - (nonZB / (Totalcells * ucnt));

// ################## Calculating Sparsity for the chosen cluster before adding implicit Data  #################//

string sqlselect;
OleDbConnection myconn = new OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=h:\\newdb.mdb");
myconn.Open();
OleDbCommand mycmd;

for (int mm = 0; mm < cntisbn; mm++)
{
sqlselect = "update booksimplicitrate set bookrating = " + isbnclusterarray[mm, 2] + " where isbn = '" +
isbnclusterarray[mm, 0] + "'";
mycmd = new OleDbCommand(sqlselect, myconn);
mycmd.ExecuteNonQuery();
}

// #########################################  update users position ##################################### //
for (int mm = 0; mm < usercnt; mm++)
{
```

```
if (usersinclusters[mm, 0] == clname)
{
sqlselect = "update booksimplicitrate set userpostion = " + mm + " where userid = " + userarray[mm] + "";
mycmd = new OleDbCommand(sqlselect, myconn);
mycmd.ExecuteNonQuery();
}
}

// ################################### update isbn position #########################################//
for (int mm = 0; mm < isbncnt; mm++)
{
sqlselect = "update booksimplicitrate set isbnpostion = " + mm + " where isbn = '" + isbnarray[mm] + "'";
mycmd = new OleDbCommand(sqlselect, myconn);
mycmd.ExecuteNonQuery();
}



// ################################### update implicit table  #########################################//
Console.Write("Cluster No. = " + clname);
Console.ReadLine();
string sql = "select userpostion, isbnpostion, bookrating from booksimplicitrate where clusterno = " + clname + ""; //
implicit feedback data
mycmd = new OleDbCommand(sql, myconn);
OleDbDataReader dr = mycmd.ExecuteReader();
// ################################### update implicit table  #########################################//

while (dr.Read())
{
usersinclusters[dr.GetInt32(0), dr.GetInt32(1) + 2] = dr.GetDouble(2);
}
dr.Close();
usersisbnrates = new double[ucnt, isbncnt + 2];
for (int q = 0; q < usercnt; q++)
{
if (usersinclusters[q, 0] == clname)
{
for (int w = 2; w < isbncnt + 2; w++)
{
usersisbnrates[qq, 0] = usersinclusters[q, 0];
usersisbnrates[qq, 1] = usersinclusters[q, 1];
usersisbnrates[qq, w] = usersinclusters[q, w];
}
qq = qq + 1;
}
}
```

```
// ##################### Calculating Sparsity for the chosen cluster after adding implicit Data ##############
double nonZA = 0;
for (int q = 0; q < ucnt; q++)
{
for (int w = 2; w < isbncnt + 2; w++)

{
if (usersisbnrates[q, w] != 0)
{
nonZA = nonZA + 1;
}
}
}
SparsityclusterImplicity = 1 - (nonZA / (Totalcells * ucnt));
/###################### Calculating Sparsity for the chosen cluster after adding implicit Data ##############

sqlselect = "update booksimplicitrate set bookrating = null, clusterno = null, userpostion = null, isbnpostion = null";
mycmd = new OleDbCommand(sqlselect, myconn);
mycmd.ExecuteNonQuery();
} // k
```

Appendix D

Distribution of users among the clusters

Tables D.1, .D.2, D.3, D.4, D.5, D.6, D.7, D.8, D.9 and D.10 show the distribution of users among the clusters k, k = 5 - 50.

Cluster #	Sparsity Level	T. User	MAE
1	98.43%	100906	0.41667
		60707	0.57847
		75591	0.74358
		97874	0.86667
		76499	0.56522
3	99.76%	7346	1
		31556	1.16667
		8067	1.25
4	98.18%	60244	0.55556
		100459	0.38095

Table D.1. No. of clusters = 5

Cluster #	Sparsity Level	T. User	MAE
0	97.55%	7346	0.53333
		8067	0.5
1	99.78%	75591	0.79464
		31556	1
		97874	1.42857
5	98.24%	60707	0.8324
7	95.10%	100906	0.33333
		100459	0.38922
9	98.76%	60244	0.57464
		76499	0.67647

Table D.2. No. of clusters = 10

Cluster #	Sparsity Level	T. User	MAE
2	98.20%	97874	0.63704
3	99.78%	60244	0.58824
		31556	1
		75591	0.75
		60707	0.83945
6	95.12%	100459	0.33333
10	98.34%	76499	0.46154
12	97.88%	100906	0.4
13	97.52%	8067	0.62054
		7346	0.46429

Table D.3. No. of clusters = 15

Cluster #	Sparsity Level	T. User	MAE
1	97.44%	7346	0.5
		100906	0.5
3	98.23%	31556	0.5883
4	95.11%	100459	0.38451
11	98.76%	60244	0.5
		75591	0.69697
16	99.78%	60707	0.87795
		76499	0.52778
		97874	0.71429
		8067	0.80556

Table D.4. No. of clusters = 20

Cluster #	Sparsity Level	T. User	MAE
3	99.77%	60707	0.96022
		100459	0.6859
		97874	0.75
4	97.51%	8067	0.6
8	98.30%	100906	0.65569
10	97.35%	7346	0.575
		75591	0.59375
15	96.21%	76499	0.36842
20	96.21%	60244	0.53186
24	96.68%	31556	0.33333

Table D.5. No. of clusters = 25

Cluster #	Sparsity Level	T. User	MAE
4	97.22%	60244	0.60294
		100906	0.57143
9	83.08%	75591	0
12	95.41%	8067	0.125
15	99.80%	100459	1
		97874	1
23	98.88%	60707	0.89573
		31556	0.725
27	94.58%	76499	0.33333
29	98.39%	7346	0.7

Table D.6. No. of clusters = 30

Cluster #	Sparsity Level	T. User	MAE
9	97.67%	100459	0.43222
11	89.64%	76499	0.38871
17	98.46%	97874	0.65891
22	97.52%	7346	0.48333
		75591	0.40258
24	99.78%	60707	0.84129
		31556	0.75655
		8067	0.68273
27	97.69%	100906	0.55714
		60244	0.51222

Table D.7. No. of clusters = 35

Cluster #	Sparsity Level	T. User	MAE
1	98.26%	60244	0.54896
		60707	0.58333
7	98.42%	100459	0.58411
12	96.15%	8067	0.35714
15	98.13%	7346	0.49856
26	99.78%	97874	0.65874
		31556	0.78294
32	97.17%	75591	0.49856
		76499	0.43333
37	95.25%	100906	0.29246

Table D.8. No. of clusters = 40

Cluster #	Sparsity Level	T. User	MAE
7	97.38	76499	0.53846
		75591	0.49877
11	98.04	97874	0.58333
21	96.06	100906	0.46667
23	98.98	7346	0.5
25	99.79	31556	0.65912
		100459	0.57143
		60244	0.59472
26	78.81	8067	0.3521
37	88.96	60707	0.45512

Table D.9. No. of clusters = 45

Cluster #	Sparsity Level	T. User	MAE
3	99.78%	7346	0.52564
		31556	0.66667
		60244	0.49825
		76499	0.57692
		100459	0.49825
		100906	0.49861
8	92.24%	60707	0.32515
11	97.5%	8067	0.46891
12	95.66%	97874	0.57407
15	98.37%	75591	0.5

Table D.10. No. of clusters = 50

References

Abdullah, C. Z. H. and N. A. Kassim (2012). Enhancing e-book selection practices in Malaysian academic libraries. Business, Engineering and Industrial Applications (ISBEIA), 2012 IEEE Symposium on.

Adeleke, A. A. and R. Olorunsola (2010). "ICT and library operations." The Electronic Library **28**(3): 453-462.

Adeniran, P. (2013). "Usage of electronic resources by undergraduates at the Redeemer's University, Nigeria." International Journal of Library and Information Science **5**(10): 319-324.

Adeniyi, D. A., Z. Wei, et al. (2014). "Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method." Applied Computing and Informatics.

Adomavicius, G. and A. Tuzhilin (2005). "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions." IEEE Trans. on Knowl. and Data Eng. **17**(6): 734-749.

Aghakhani, N., A. S. Najar, et al. (2010). Supporting research collaboration through web based personal digital library. Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on.

Agrawal, R. and R. Srikant (1994). Fast Algorithms for Mining Association Rules in Large Databases. Proceedings of the 20th International Conference on Very Large Data Bases, Morgan Kaufmann Publishers Inc.**:** 487-499.

Agrawal, R. and R. Srikant (1995). Mining Sequential Patterns. Proceedings of the Eleventh International Conference on Data Engineering, IEEE Computer Society**:** 3-14.

AGUILAR, J. (2009). "A Web Mining System." WSEAS TRANSACTIONS on INFORMATION SCIENCE and APPLICATIONS **6**(9): 1523-1532.

Ahn, Y. H. Cho, et al. (2004). A comparative evaluation of hybrid product recommendation procedures for Web retailers. 9th Asia-Pacific Decision Science Institute Conference (APDSI). Seoul.

Ahn, H. J. (2008). "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem." Information Sciences **178**(1): 37-51.

Ahn, H. J., H. Kang, et al. (2010). "Selecting a small number of products for effective user profiling in collaborative filtering." Expert Systems with Applications **37**(4): 3055-3062.

Aijuan, D. and W. Baoying (2008). Domain-Based Recommendation and Retrieval of Relevant Materials in E-learning. Semantic Computing and Applications, 2008. IWSCA '08. IEEE International Workshop on.

Akbar, M., C. A. Shaffer, et al. (2014). Recommendation based on deduced social networks in an educational digital library. Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries. London, United Kingdom, IEEE Press**:** 29-38.

Akbar, M., C. A. Shaffer, et al. (2014). Recommendation based on Deduced Social Networks in an educational digital library. Digital Libraries (JCDL), 2014 IEEE/ACM Joint Conference on.

Alam, S. (2011). Intelligent web usage clustering based recommender system. Proceedings of the fifth ACM conference on Recommender systems. Chicago, Illinois, USA, ACM**:** 367-370.

AlMurtadha, Y., N. B. Sulaiman, et al. (2011). "IPACT: Improved Web Page Recommendation System Using Profile Aggregation Based On Clustering of Transactions " American Journal of Applied Sciences **8**(3): 277-283.

Amatriain, X., A. Jaimes, et al. (2011). Data Mining Methods for Recommender Systems. Recommender Systems Handbook. F. Ricci, L. Rokach, B. Shapira and P. B. Kantor. Boston, MA, Springer US**:** 39-71.

Ambayo, J. A. (2010). A Conceptual data mining model (DMM) used in Selective Dissemination of Information (SDI) : case study - Strathmore University Library. Faculty of Information Technology, Strathmore University, Kenya. **MSc in Computer Based Information Systems:** 50.

Anaraki, L. N. and A. Heidari (2011). Knowledge management process in digital age: Proposing a model for implementing e-learning through digital libraries. Application of Information and Communication Technologies (AICT), 2011 5th International Conference on.

Andonie, R., J. E. Russo, et al. (2007). "Crossing the Rubicon for An Intelligent Advisor." International Journal of Computers, Communications & Control (IJCCC) **2**(1): 5-16.

Apte, C., B. Liu, et al. (2002). "Business applications of data mining." Commun. ACM **45**(8): 49-53.

Arivazhagan and R. Pragaladan (2015). "Re-Adapted Apriori Algorithm in E-Commerce Proposal Coordination." International Journal of Innovative Research in Computer and Communication Engineering **3**(8): 7329-7336.

Avancini, H. and U. Straccia (2005). "User recommendation for collaborative and personalised digital archives." Int. J. Web Based Communities **1**(2): 163-175.

Azam, I., S. J. Sohrawardi, et al. (2013). Bibliomining on North South University library data. Digital Information Management (ICDIM), 2013 Eighth International Conference on.

Balabanovi, M. and Y. Shoham (1997). "Fab: content-based, collaborative recommendation." Commun. ACM **40**(3): 66-72.

Banerjee, A. and J. Ghosh (2001). Clickstream Clustering using Weighted Longest Common Subsequences. In Proceedings of the Web Mining Workshop at the 1st SIAM Conference on Data Mining, Chicago.

Bargah, P. and N. Mishra (2016). "Solving Sparsity Problem in Movie Based Recommendation System." Advances in Image and Video Processing **4**(3).

Barker, L. J. (2009). Science teachers' use of online resources and the digital library for Earth system education. Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries. Austin, TX, USA, ACM**:** 1-10.

Bazillion, R. J. (2001). "Academic Libraries in the Digital Revolution." Educause Quarterly **24**(1): 51-55.

Beel, J. (2015). Towards Effective Research-Paper Recommender Systems and User Modeling based on Mind Maps. faculty Of computer science, magdeburg university

PhD: 307.

Bell, R. M. and Y. Koren (2007). Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights. Proceedings of the 2007 Seventh IEEE International Conference on Data Mining, IEEE Computer Society**:** 43-52.

Bertino, E., E. Ferrari, et al. (2001). An Access Control Mechanism for Large Scale Data Dissemination Systems. Eleventh International Workshop on Research Issues in

Data Engineering on Document Management for Data Intensive Business and Scientific Applications, IEEE Computer Society: 43-50.

Bhide, A. M., Y. J. Heung, et al. (2007). Research Library: A New Look of Academic Digital Libraries. Proceedings of the Second International Conference on Internet and Web Applications and Services, IEEE Computer Society: 57.

Bhide, A. M., H. Yoo Jae, et al. (2007). Research Library: A New Look of Academic Digital Libraries. Internet and Web Applications and Services, 2007. ICIW '07. Second International Conference on.

Bidart, R., A. C. M. Pereira, et al. (2014). A Characterization of Access Profiles and Navigation in E-Commerce: A Tourism Application. Proceedings of the 20th Brazilian Symposium on Multimedia and the Web. Jo&#227;o Pessoa, Brazil, ACM: 17-20.

Billsus, D. and M. J. Pazzani (1998). Learning Collaborative Information Filters. Proceedings of the Fifteenth International Conference on Machine Learning, Morgan Kaufmann Publishers Inc.: 46-54.

Board, E. A. (2011). Redefining the Academic Library: Managing the Migration to Digital Information Services. Washington, DC, University Leadership Council.

Bobadilla, J., F. Ortega, et al. (2012). "A collaborative filtering approach to mitigate the new user cold start problem." Knowledge-Based Systems **26**: 225-238.

Bobadilla, J., F. Ortega, et al. (2013). "Recommender systems survey." Know.-Based Syst. **46**: 109-132.

Bobadilla, J., F. Serradilla, et al. (2009). "Collaborative filtering adapted to recommender systems of e-learning." Knowledge-Based Systems **22**(4): 261-265.

Boddu, S. B., V. P. K. Anne, et al. (2010). Knowledge Discovery and Retrieval on World Wide Web Using Web Structure Mining. Mathematical/Analytical Modelling and Computer Simulation (AMS), 2010 Fourth Asia International Conference on.

Bonnin, G. and D. Jannach (2013). A Comparison of Playlist Generation Strategies for Music Recommendation and a New Baseline Scheme. Twenty-Seventh AAAI Conference on Artificial Intelligence. Washington USA, AAAI Publications.

Borhani-Fard, Z. M., Behrouz; Alinejad-Rokny, Hamid (2013). "Applying Clustering Approach in Blog Recommendation." Journal of Emerging Technologies in Web Intelligence **5**(3).

Bradley, K., R. Rafter, et al. (2000). Case-Based User Profiling for Content Personalisation. Adaptive Hypermedia and Adaptive Web-Based Systems: International Conference, AH 2000 Trento, Italy, August 28–30, 2000 Proceedings. P. Brusilovsky, O. Stock and C. Strapparava. Berlin, Heidelberg, Springer Berlin Heidelberg**:** 62-72.

Brangier, E., J. Dinet, et al. (2009). The 7 Basic Functions of a Digital Library - Analysis of Focus Groups about the Usefulness of a Thematic Digital Library on the History of European Integration. Human Interface and the Management of Information. Designing Information Environments. M. Smith and G. Salvendy, Springer Berlin Heidelberg. **5617:** 345-354.

Breese, J. S., D. Heckerman, et al. (1998). Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. Madison, Wisconsin, Morgan Kaufmann Publishers Inc.**:** 43-52.

Brigitte, T., A. Marie-Aude, et al. (2008). Web Usage Mining for Ontology Management. Data Mining with Ontologies: Implementations, Findings, and Frameworks. N. Hector Oscar, C. Sandra Elizabeth Gonzalez and X. Daniel Hugo. Hershey, PA, USA, IGI Global**:** 37-64.

Brusilovsky, P., R. Farzan, et al. (2005). Comprehensive personalized information access in an educational digital library. Digital Libraries, 2005. JCDL '05. Proceedings of the 5th ACM/IEEE-CS Joint Conference on.

Buchanan, G., D. Bainbridge, et al. (2005). A new framework for building digital library collections. Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries. Denver, CO, USA, ACM**:** 23-31.

Bundza, M. (2009). "Work of the Web Weavers: Web Development in Academic Libraries." Journal of Web Librarianship **3**(3): 24.

Burke, R. (2002). "Hybrid Recommender Systems: Survey and Experiments." User Modeling and User-Adapted Interaction **12**(4): 331-370.

Burke, R. (2007). Hybrid web recommender systems. The adaptive web. B. Peter, K. Alfred and N. Wolfgang, Springer-Verlag**:** 377-408.

Cadez, I., D. Heckerman, et al. (2003). "Model-Based Clustering and Visualization of Navigation Patterns on a Web Site." Data Mining and Knowledge Discovery **7**(4): 399-424.

Cadez, I. V., P. Smyth, et al. (2001). Probabilistic modeling of transaction data with applications to profiling, visualization, and prediction. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. San Francisco, California, ACM**:** 37-46.

Carlson, B. and S. Reidy (2004). "Effective access: teachers' use of digital resources (research in progress)." OCLC Systems & Services: International digital library perspectives **20**(2): 65-70.

Castro-Schez, J. J., R. Miguel, et al. (2011). "A highly adaptive recommender system based on fuzzy logic for B2C e-commerce portals." Expert Systems with Applications **38**(3): 2441-2454.

Chao, D. L., J. Balthrop, et al. (2005). Adaptive radio: achieving consensus using negative preferences. Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work. Sanibel Island, Florida, USA, ACM**:** 120-123.

Chen, C. C. and A. P. Chen (2007). "Using data mining technology to provide a recommendation service in the digital library." The Electronic Library **25**(6): 711-724.

Chen, K. and L. Liu (2003). Cluster rendering of skewed datasets via visualization. Proceedings of the 2003 ACM symposium on Applied computing. Melbourne, Florida, ACM**:** 909-916.

Chen, R.-S., Y.-S. Tsai, et al. (2008). "Using data mining to provide recommendation service." WSEAS Trans. Info. Sci. and App. **5**(4): 459-474.

Chen, Y., C. Wu, et al. (2011). "Solving the Sparsity Problem in Recommender Systems Using Association Retrieval." Journal of Computers **6**(9): 1896-1902.

Chen, Y., C. Wu, et al. (2011). "Solving the Sparsity Problem in Recommender Systems Using Association Retrieval." JOURNAL OF COMPUTERS **6**(9): 1896-1902.

Chen, Z. (2011). Exploring new trends of university libraries by SPSS cluster analysis method &#x2014; Take Wuhan University of Technology as an example. Product Innovation Management (ICPIM), 2011 6th International Conference on.

Chi-Jen, W., C. Jen-Ming, et al. (2011). Using Web-Mining for Academic Measurement and Scholar Recommendation in Expert Finding System. Web Intelligence and Intelligent Agent Technology (WI-IAT), 2011 IEEE/WIC/ACM International Conference on.

Cho, Y. H. and J. K. Kim (2004). "Application of Web usage mining and product taxonomy to collaborative recommendations in e-commerce." Expert Systems with Applications **26**(2): 233-246.

Choi, J. and L. Geehyuk (2009). New Techniques for Data Preprocessing Based on Usage Logs for Efficient Web User Profiling at Client Side. Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on.

Christidis, K. and G. Mentzas (2013). "A topic-based recommender system for electronic marketplace platforms." Expert Systems with Applications **40**(11): 4370-4379.

Claypool, M., P. Le, et al. (2001). Implicit interest indicators. Proceedings of the 6th international conference on Intelligent user interfaces. Santa Fe, New Mexico, USA, ACM**:** 33-40.

Cooley, R., B. Mobasher, et al. (1997). Web mining: information and pattern discovery on the World Wide Web. Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on.

Cooley, R., B. Mobasher, et al. (1999). "Data Preparation for Mining World Wide Web Browsing Patterns." Knowl. Inf. Syst. **1**(1): 5-32.

Crespo, R. G., O. S. Martínez, et al. (2011). "Recommendation System based on user interaction data applied to intelligent electronic books." Computers in Human Behavior **27**(4): 1445-1449.

Dakhel, G. M. and M. Mahdavi (2011). A new collaborative filtering algorithm using K-means clustering and neighbors' voting. Hybrid Intelligent Systems (HIS), 2011 11th International Conference on.

Dakhel, G. M. and M. Mahdavi (2011). A new collaborative filtering algorithm using K-means clustering and neighbors' voting. 2011 11th International Conference on Hybrid Intelligent Systems (HIS).

Demiriz, A. (2004). "Enhancing Product Recommender Systems on Sparse Binary Data." Data Min. Knowl. Discov. **9**(2): 147-170.

Demovic, L., E. Fritscher, et al. (2013). Movie Recommendation Based on Graph Traversal Algorithms. Database and Expert Systems Applications (DEXA), 2013 24th International Workshop on.

Denis Parra , A. K., Idil Yavuz , Xavier Amatriain (2011). Implicit feedback recommendation via implicit-to-explicit ordinal logistic regression mapping. In Proceedings of the CARS-2011.

Dez, F., J. E. Chavarriaga, et al. (2010). Movie recommendations based in explicit and implicit features extracted from the <i>Filmtipset</i> dataset. Proceedings of the Workshop on Context-Aware Movie Recommendation. Barcelona, Spain, ACM**:** 45-52.

Diallo, A. and L. Liwen (2011). Management of an Academic E-Library Project. Information Management, Innovation Management and Industrial Engineering (ICIII), 2011 International Conference on.

Dianjun, X. and S. Min (2011). Research on application of digital library based on association rules. Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on.

Dingming, W., Z. Dongyan, et al. (2008). An Adaptive User Profile Based on Memory Model. Web-Age Information Management, 2008. WAIM '08. The Ninth International Conference on.

Dinkins, D. (2011). "Allocating Academic Library Budgets: Adapting Historical Data Models at One University Library." Collection Management **36**(2): 119-130.

Dixit, D., J. Gadge, et al. (2010). "Automatic Recommendation of Web Pages in Web Usage Mining." International Journal of Managing Information Technology (IJMIT) **2**(3).

Dollah, W. Ab, et al. (2006). Digital reference services in selected public academic libraries in Malaysia: A case study. Asia-Pacific Conference on Library & Information Education & Practice 2006(A-LIEP 2006), Singapore.

Dollah, W. A. K. W. (2008). Determining the Effectiveness of Digital Reference Services in Selected Academic Libraries in Malaysia. Faculty of Computer Science and Information Technology. Kuala Lumpur, University of Malaya. **Degree of Doctor of Philosophy:** 269.

Dong, L., Y. Nie, et al. (2006). Research and Implementation of a Personalized Recommendation System. Digital Libraries: Achievements, Challenges and Opportunities. S. Sugimoto, J. Hunter, A. Rauber and A. Morishima, Springer Berlin Heidelberg. **4312:** 183-191.

Dong, R., L. Tokarchuk, et al. (2009). Digging Friendship: Paper Recommendation in Social Network. Proceedings of Networking & Electronic Commerce Research Conference (NAEC 2009).

Donnellan, D. and C. Pahl (2002). Data Mining Technology for the Evaluation of Web-based Teaching and Learning Systems. World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2002. M. Driscoll and T. C. Reeves. Montreal, Canada, Association for the Advancement of Computing in Education (AACE)**:** 747-752.

Dorigo, M. and L. M. Gambardella (1997). "Ant colony system: a cooperative learning approach to the traveling salesman problem." Evolutionary Computation, IEEE Transactions on **1**(1): 53-66.

Edmunds, A. and A. Morris (2000). "The problem of information overload in business organisations: a review of the literature." International Journal of Information Management **20**(1): 17-28.

Ekstrand, M. D., J. T. Riedl, et al. (2011). "Collaborative Filtering Recommender Systems." Found. Trends Hum.-Comput. Interact. **4**(2): 81-173.

Eltahir, M. A. and A. F. A. Dafa-Alla (2013). Extracting knowledge from web server logs using web usage mining. Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on.

Esmailian, P. and M. Jalili (2015). Purchase Prediction and Item Suggestion based on HTTP sessions in absence of User Information. Proceedings of the 2015 International ACM Recommender Systems Challenge. Vienna, Austria, ACM**:** 1-4.

Esslimani, I., A. Brun, et al. (2009). A collaborative filtering approach combining clustering and navigational based correlations. 5th International Conference on Web Information Systems and Technologies - WEBIST 2009, Lisbonne, Portugal.

Esteban, B., Á. Tejeda-Lorente, et al. (2014). Aiding in the Treatment of Low Back Pain by a Fuzzy Linguistic Web System. Rough Sets and Current Trends in Computing. C. Cornelis, M. Kryszkiewicz, D. Ślęzaket al, Springer International Publishing. **8536:** 250-261.

Fan, W., H. Ya-Han, et al. (2011). Decision Support in Library Book Acquisition: A Social Computing-Based Approach. e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on.

Fox, S., K. Karnawat, et al. (2005). "Evaluating implicit measures to improve web search." ACM Trans. Inf. Syst. **23**(2): 147-168.

Freund, Y., R. Iyer, et al. (2003). "An efficient boosting algorithm for combining preferences." J. Mach. Learn. Res. **4**: 933-969.

Fuchs, M. and M. Zanker (2012). Multi-criteria Ratings for Recommender Systems: An Empirical Analysis in the Tourism Domain. Proceedings of the 13th International Conference on Electronic Commerce and Web Technologies, Vienna, Austria, Springer Berlin Heidelberg.

Furner, J. (2002). "On Recommending." Journal of the American Society for Information Science and Technology.

Gao, F., C. Xing, et al. (2007). An Effective Algorithm for Dimensional Reduction in Collaborative Filtering. Looking Back 10 Years and Forging New Frontiers. D. H.-L. Goh, T. H. Cao, I. T. Sølvberg and E. Rasmussen. Berlin, Heidelberg, Springer Berlin Heidelberg**:** 75-84.

Gauch, S., M. Speretta, et al. (2007). User profiles for personalized information access. The adaptive web. B. Peter, K. Alfred and N. Wolfgang, Springer-Verlag**:** 54-89.

GIBSON, I. E. (2001). Data mining analysis of digital library database usage patterns as a tool facilitating efficient user navigation. College of Communications. TUSCALOOSA, ALABAMA, The University of Alabama. **Doctor of Philosophy:** 121.

Gong, S. (2010). "A Collaborative Filtering Recommendation Algorithm Based on User Clustering and Item Clustering." JOURNAL OF SOFTWARE **5**(7): 745-752.

Grace1, L. K. J., V.Maheswari, et al. (2011). "ANALYSIS OF WEB LOGS AND WEB USER IN WEB MINING." International Journal of Network Security & Its Applications (IJNSA) **3**(1): 99-110.

Guozheng, H. and C. Rongqiu (2007). E-Enterprise and E-Management Concept and Process Model Research. Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on.

Gupta, P., A. Sharma, et al. (2014). "Data Harvesting through Web Mining: A Survey." The International Journal Of Engineering And Science (IJES) **3**(4): 21-27.

Guy, I., N. Zwerdling, et al. (2010). Social media recommendation based on people and tags. Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. Geneva, Switzerland, ACM**:** 194-201.

Han, L. and A. Goulding (2003). "Information and reference services in the digital library." Information Services and Use **23** 251-262.

Herlocker, J., Konstan, J.A. & Riedl, J. (2002). "An Empirical Analysis of Design Choices in Neighborhood-Based Collaborative Filtering Algorithms." Information Retrieval **5**(4): 287–310.

Herlocker, J. L., J. A. Konstan, et al. (1999). An algorithmic framework for performing collaborative filtering. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. Berkeley, California, USA, ACM**:** 230-237.

Herlocker, J. L., J. A. Konstan, et al. (2004). "Evaluating collaborative filtering recommender systems." ACM Trans. Inf. Syst. **22**(1): 5-53.

Hernando, A., Jes, et al. (2016). "A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model." Know.-Based Syst. **97**(C): 188-202.

Hossain, M. and R. M. Rahman (2014). Budget allocation model for the academic library acquisition using data mining technique. Computer and Information Technology (ICCIT), 2013 16th International Conference on.

Hu, Y., Y. Koren, et al. (2008). Collaborative Filtering for Implicit Feedback Datasets. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, IEEE Computer Society**:** 263-272.

Huaxin, C. and H. Qian (2013). Research on intelligent recommended algorithms of personalized digital library. Conference Anthology, IEEE.

Hwang, S. Y. and S. M. Chuang (2004). "Combining article content and Web usage for literature recommendation in digital libraries." Online Information Review **28**(4): 260-272.

Hwang, S. Y., W. C. Hsiung, et al. (2003). "A prototype WWW literature recommendation system for digital libraries." Online Information Review **27**(3): 169-182.

Im, I. and A. Hars (2007). "Does a one-size recommendation system fit all? the effectiveness of collaborative filtering based recommendation systems across different domains and search modes." ACM Trans. Inf. Syst. **26**(1): 4.

Impagliazzo, J., J. A. N. Lee, et al. (2003). Using the NSF digital library to enhance your teaching. Frontiers in Education, 2003. FIE 2003 33rd Annual.

Isinkaye, F. O., Y. O. Folajimi, et al. (2015). "Recommendation systems: Principles, methods and evaluation." Egyptian Informatics Journal **16**(3): 261-273.

Iváncsy, R. and S. Juhász (2007). "Analysis of Web User Identification Methods " International Journal of Computer, Electrical, Automation, Control and Information Engineering **1**(10): 2995-3002.

Jalali, M., N. Mustapha, et al. (2010). "WebPUM: A Web-based recommendation system to predict user future movements." Expert Systems with Applications **37**(9): 6201-6212.

Jange, S. (2015). Innovative services and practices in academic libraries. Emerging Trends and Technologies in Libraries and Information Services (ETTLIS), 2015 4th International Symposium on.

Jawaheer, G., P. Weller, et al. (2014). "Modeling User Preferences in Recommender Systems: A Classification Framework for Explicit and Implicit User Feedback." ACM Trans. Interact. Intell. Syst. **4**(2): 1-26.

Jespersen, S., J. Thorhauge, et al. (2002). A Hybrid Approach to Web Usage Mining. Data Warehousing and Knowledge Discovery. Y. Kambayashi, W. Winiwarter and M. Arikawa, Springer Berlin Heidelberg. **2454:** 73-82.

Jie, Y., Z. Haihong, et al. (2012). A Method of Discovering Collaborative Users Based on Psychological Model in Academic Recommendation. Computer and Information Technology (CIT), 2012 IEEE 12th International Conference on.

Jinmook Kim, D. W. Oard, et al. (2000). Using implicit feedback for user modeling in Internet and Intranet searching.**:** 1-21.

Joachims, T. (2002). Optimizing search engines using clickthrough data. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. Edmonton, Alberta, Canada, ACM**:** 133-142.

Joachims, T., L. Granka, et al. (2005). Accurately interpreting clickthrough data as implicit feedback. Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. Salvador, Brazil, ACM**:** 154-161.

Joshi1, R. C. and R. S. Paswan (2013). "A Survey Paper onClustering-based Collaborative Filtering Approach to Generate Recommendations." International Journal of Science and Research (IJSR) **4**(1): 1395-1398.

Juan, Z., D. Kejun, et al. (2009). E- Scholar: Improving academic search through combining metasearch with entity extraction. Information, Computing and Telecommunication, 2009. YC-ICT '09. IEEE Youth Conference on.

Jung, S. (2007). Designing and understanding information retrieval systems using collaborative filtering in an academic library environment, Oregon State University**:** 109.

Kadir, R., W. A. K. Dollah, et al. (2009). A User-based measure in evaluating academic digital library. International Conference on Academic Libraries (ICAL 2009). University of Delhi (North Campus), Delhi, India.

Kadir, R. A., S. A. Rahman, et al. (2014). Demographic Factors and Awareness of Academic Digital Libraries at Higher Learning Institutions. The 24th IBIMA conference on Crafting Global Competitive Economies: 2020 Vision Strategic Planning & Smart Implementation, Milan, Itely.

Kanimozhi, S. (2011). "Effective Constraint based Clustering Approach for Collaborative Filtering Recommendation using Social Network Analysis." Bonfring International Journal of Data Mining **1**(Special Issue): 12-17.

Kantardzic, M. (2002). Data Mining: Concepts, Models, Methods and Algorithms, John Wiley \&amp; Sons, Inc.

Kao, S. C., H. C. Chang, et al. (2003). "Decision support for the academic library acquisition budget allocation via circulation database mining." Information Processing & Management **39**(1): 133-147.

Ke, H.-R., R. Kwakkelaar, et al. (2002). "Exploring behavior of E-journal users in science and technology: Transaction log analysis of Elsevier's ScienceDirect OnSite in Taiwan." Library & Information Science Research **24**(3): 265-291.

Kelly, D. and N. J. Belkin (2004). Display time as implicit feedback: understanding task effects. Proceedings of the 27th annual international ACM SIGIR conference on

Research and development in information retrieval. Sheffield, United Kingdom, ACM**:** 377-384.

Kelly, D. and J. Teevan (2003). "Implicit feedback for inferring user preference: a bibliography." SIGIR Forum **37**(2): 18-28.

Keralapura, M. (2009). "Technology and customer expectation in academic libraries: A special reference to technical/management libraries in Karnataka." The International Information & Library Review **41**(3): 184-195.

Kim, K.-j. and H. Ahn (2008). "A recommender system using GA K-means clustering in an online shopping market." Expert Systems with Applications **34**(2): 1200-1209.

Kim, Y. S., B.-J. Yum, et al. (2005). "Development of a recommender system based on navigational and behavioral patterns of customers in e-commerce sites." Expert Syst. Appl. **28**(2): 381-393.

Konstan, J. A., B. N. Miller, et al. (1997). "GroupLens: applying collaborative filtering to Usenet news." Commun. ACM **40**(3): 77-87.

Konstan, J. A., J. D. Walker, et al. (2014). Teaching recommender systems at large scale: evaluation and lessons learned from a hybrid MOOC. Proceedings of the first ACM conference on Learning @ scale conference. Atlanta, Georgia, USA, ACM**:** 61-70.

Koren, Y. (2010). "Factor in the neighbors: Scalable and accurate collaborative filtering." ACM Trans. Knowl. Discov. Data **4**(1): 1-24.

Koren, Y., R. Bell, et al. (2009). "Matrix Factorization Techniques for Recommender Systems." Computer **42**(8): 30-37.

Kosala, R. and H. Blockeel (2000). "Web mining research: a survey." SIGKDD Explor. Newsl. **2**(1): 1-15.

Kouser, K. and Sunita (2013). "A comparative study of K Means Algorithm by Different Distance Measures." International Journal of Innovative Research in Computer and Communication Engineering **1**(9): 2443-2447.

Kovacevic, A., V. Devedzic, et al. (2010). "Using data mining to improve digital library services." The Electronic Library **28**(6): 829-843.

Krishnamurthy, V. and R. Balasubramani (2014). An Association Rule Mining Approach for Libraries to Analyse User Interest. Intelligent Computing Applications (ICICA), 2014 International Conference on.

Kun, M., L. Tingting, et al. (2014). Hybrid parallel approach for personalized literature recommendation system. Computational Aspects of Social Networks (CASoN), 2014 6th International Conference on.

Kuo, W. T., Y. C. Wang, et al. (2015). Contextual restaurant recommendation utilizing implicit feedback. 2015 24th Wireless and Optical Communication Conference (WOCC).

Kuzelewska, U. (2014). "Clustering Algorithms in Hybrid Recommender System on MovieLens Data." Studies in Logic, Grammar and Rhetoric. **37**(1): 125–139.

Kveton, B. and S. Berkovsky (2015). Minimal Interaction Search in Recommender Systems. Proceedings of the 20th International Conference on Intelligent User Interfaces. Atlanta, Georgia, USA, ACM**:** 236-246.

Lee, D. H. and P. Brusilovsky (2009). Reinforcing Recommendation Using Implicit Negative Feedback. User Modeling, Adaptation, and Personalization: 17th International Conference, UMAP 2009, formerly UM and AH, Trento, Italy, June 22-26, 2009. Proceedings. G.-J. Houben, G. McCalla, F. Pianesi and M. Zancanaro. Berlin, Heidelberg, Springer Berlin Heidelberg**:** 422-427.

Lee, S. K., Y. H. Cho, et al. (2010). "Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations." Information Sciences **180**(11): 2142-2155.

Lee, Y. (2015). RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING. The Faculty of the Department of Computer Science, San José State University. **Master**.

Lei, M. and Y. Lingshui (2010). Research on digital library knowledge organization methods based-on semantic grid. Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on.

Lekakos, G. and G. M. Giaglis (2006). "Improving the prediction accuracy of recommendation algorithms: Approaches anchored on human factors." Interact. Comput. **18**(3): 410-431.

Lemire, D., H. Boley, et al. (2005). "Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation."

Leung, C. W.-k., S. C.-f. Chan, et al. (2008). "An empirical study of a cross-level association rule mining approach to cold-start recommendations." Knowledge-Based Systems **21**(7): 515-529.

Li, H., Y. Gu, et al. (2009). Review of Digital Library Book Recommendation Models. Available at SSRN: http://ssrn.com/abstract=1513415

Li, J. and P. Chen (2008). The application of Association rule in Library system. Knowledge Acquisition and Modeling Workshop, 2008. KAM Workshop 2008. IEEE International Symposium on.

Li, Q. and B. M. Kim (2003). Clustering Approach for Hybrid Recommender System. Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence, IEEE Computer Society**:** 33.

Li, Q., J. Wang, et al. (2010). "User comments for news recommendation in forum-based social media." Information Sciences **180**(24): 4929-4939.

Liang, T.-P. (2008). "Recommendation systems for decision support: An editorial introduction." Decision Support Systems **45**(3): 385-386.

Lichtenstein, S. and P. Slovic (2006). The Construction of Preference, Cambridge University Press.

Lika, B., K. Kolomvatsos, et al. (2014). "Facing the cold start problem in recommender systems." Expert Systems with Applications **41**(4, Part 2): 2065-2073.

Lin, C., R. Xie, et al. (2014). "Personalized news recommendation via implicit social experts." Information Sciences **254**: 1-18.

Lina, J. and M. Zhiyong (2013). "The Application of Book Intelligent Recommendation Based on the Association Rule Mining of Clementine " Journal of Software Engineering and Application **6**: 30-33

Linden, G., B. Smith, et al. (2003). "Amazon.com recommendations: item-to-item collaborative filtering." Internet Computing, IEEE **7**(1): 76-80.

Linghui, G. (2010). On construction of knowledge management in university digital libraries. Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on.

Liu, N. N., L. He, et al. (2013). "Social temporal collaborative ranking for context aware movie recommendation." ACM Trans. Intell. Syst. Technol. **4**(1): 1-26.

Lomotey, R. K. and R. Deters (2014). Towards Knowledge Discovery in Big Data. Service Oriented System Engineering (SOSE), 2014 IEEE 8th International Symposium on.

Lopes, P. and B. Roy (2014). "Recommendation System using Web Usage Mining for users of E-commerce site." International Journal of Engineering Research & Technology **3**(7).

Lops, P., M. de Gemmis, et al. (2011). Content-based Recommender Systems: State of the Art and Trends. Recommender Systems Handbook. F. Ricci, L. Rokach, B. Shapira and P. B. Kantor, Springer US**:** 73-105.

Lytras, M. and P. O. d. Pablos (2011). "Software technologies in knowledge society." Journal of Universal Computer Science **17**(9): 1219-1221.

Ma, L. and J. Xiao (2010). The study on customer-oriented solutions for college library services. Computer and Communication Technologies in Agriculture Engineering (CCTAE), 2010 International Conference On.

Mabroukeh, N. R. and C. I. Ezeife (2010). "A taxonomy of sequential pattern mining algorithms." ACM Comput. Surv. **43**(1): 1-41.

Mahajan, D. S., P. Pawar, et al. (2014). "Analysis of Large Web Sequences using AprioriAll_Set Algorithm." International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) **3**(2).

Malik, S. K. and S. A. M. Rizvi (2011). Information Extraction Using Web Usage Mining, Web Scrapping and Semantic Annotation. Computational Intelligence and Communication Networks (CICN), 2011 International Conference on.

Malinowski, J., T. Weitzel, et al. (2008). "Decision support for team staffing: An automated relational recommendation approach." Decision Support Systems **45**(3): 429-447.

Manh Cuong Pham, Y. C., Ralf Klamma, Matthias Jarke (2011). "A clustering approach for collaborative filtering recommendation using social network analysis." Journal of Universal Computer Science J.UCS **17**(4).

Maull, K. E., M. G. Saldivar, et al. (2010). Online curriculum planning behavior of teachers. The Third International Conference on Educational Data Mining (EDM2010). Pittsburg, PA, USA.

Mayega, S. (2008). LIBRARY INFORMATION SERVICES IN THE DIGITAL AGE. Fourth Shanghai International Library Forum (SILF 2008). Shanghai (China).

McGrath, O. G. (2008). Insights and surprises from usage patterns: some benefits of data mining in academic online systems. Proceedings of the 36th annual ACM SIGUCCS fall conference: moving mountains, blazing trails. Portland, OR, USA, ACM**:** 59-64.

McNally, K., M. P. O, et al. (2011). "A Case Study of Collaboration and Reputation in Social Web Search." ACM Trans. Intell. Syst. Technol. **3**(1): 1-29.

Meghabghab, G. and A. Kandel (2008). Search Engines, Link Analysis, and User's Web Behavior: A Unifying Web Mining Approach, Springer Publishing Company, Incorporated.

Meyyappan, N., S. Foo, et al. (2004). "Design and evaluation of a task-based digital library for the academic community." Journal of Documentation **60**(4): 449-475.

Miller, B. N., J. A. Konstan, et al. (2004). "PocketLens: Toward a personal recommender system." ACM Trans. Inf. Syst. **22**(3): 437-476.

Mobasher, B. (2007). Data mining for web personalization. The adaptive web. B. Peter, K. Alfred and N. Wolfgang, Springer-Verlag**:** 90-135.

Mobasher, B., H. Dai, et al. (2002). "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization." Data Min. Knowl. Discov. **6**(1): 61-82.

Mohd Shuib, N. L., N. Abdullah, et al. (2010). The use of information retrieval tools: A study of computer science postgraduate students. Science and Social Research (CSSR), 2010 International Conference on.

Mönnich, M. and M. Spiering (2008). Adding Value to the Library Catalog by Implementing a Recommendation System. D-Lib Magazine.

Montaner, M., B. L\, et al. (2003). "A Taxonomy of Recommender Agents on theInternet." Artif. Intell. Rev. **19**(4): 285-330.

Morales-del-Castillo, J. M., R. Pedraza-Jiménez, et al. (2009). "A Semantic Model of Selective Dissemination of Information for Digital Libraries." Information Technology and Libraries **28**(1).

More, N. and K. J. Somaiya (2014). "Recommendation of Books Using Improved Apriori Algorithm." International Journal for Innovative Research in Science & Technology - IJIRST **1**(4): 80-82.

Mulvenna, M. D., S. S. Anand, et al. (2000). "Personalization on the Net using Web mining: introduction." Commun. ACM **43**(8): 122-125.

Muralidhar, N., H. Rangwala, et al. (2015). Recommending Temporally Relevant News Content from Implicit Feedback Data. 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI).

Mustafa, A. S. and Y. S. Kumaraswamy (2014). Data mining algorithms for Web-services classification. Contemporary Computing and Informatics (IC3I), 2014 International Conference on.

Naak, A., H. Hage, et al. (2009). A multi-criteria collaborative filtering approach for research paper recommendation in papyrus. 4th International Conference MCETECH.

Nassar, O. A. and N. A. Al Saiyd (2013). The integrating between web usage mining and data mining techniques. Computer Science and Information Technology (CSIT), 2013 5th International Conference on.

Nelson, M. R. (1994). "We have the information you want, but getting it will cost you!: held hostage by information overload." Crossroads **1**(1): 11-15.

Netcraft (2015). "February 2015 Web Server Survey." from http://news.netcraft.com/archives/category/web-server-survey/.

Neumann, J. and H. Sayyadi (2015). Recommendations for Live TV. Proceedings of the 9th ACM Conference on Recommender Systems. Vienna, Austria, ACM**:** 228-228.

Nguyen, T.-T. and P.-K. Nguyen (2012). A New Approach for Problem of Sequential Pattern Mining. Computational Collective Intelligence. Technologies and Applications. N.-T. Nguyen, K. Hoang and P. Jędrzejowicz, Springer Berlin Heidelberg. **7653:** 51-60.

Nichols, D. M., D. Bainbridge, et al. (2006). Learning by building digital libraries. Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries. Chapel Hill, NC, USA, ACM**:** 185-186.

Nicholson, S. (2003). "The Bibliomining Process: Data Warehousing and Data Mining for Library Decision-Making." Information Technology and Libraries **22**(4).

Nicholson, S. (2006). "The basis for bibliomining: Frameworks for bringing together usage-based data mining and bibliometrics through data warehousing in digital library services." Information Processing & Management **42**(3): 785-804.

Nov, O. and O. Arazy (2015). Asymmetric Recommendations: The Interacting Effects of Social Ratings? Direction and Strength on Users' Ratings. Proceedings of the 9th ACM Conference on Recommender Systems. Vienna, Austria, ACM**:** 249-252.

Oard, D. and J. Kim (1998). Implicit Feedback for Recommender Systems. in Proceedings of the AAAI Workshop on Recommender Systems.

Oard, D. and J. Kim (2001). "Modeling Information Content Using Observable Behavior." Proceedings of the ASIST Annual Meeting **38**: 481.

Odongo, T. A. M. (2011). AN ASSESSMENT OF ICT ADOPTION IN KENYAN ACADEMIC LIBRARIES- A CASE STUDY OF UNIVERSITY OF NAIROBI LIBRARIES. School of Business, University of Nairobi. **Degree of Masters of Business:** 64.

Ogunsola, L. A. (2011). "The Next Step in Librarianship: Is The Traditional Library Dead?" Library Philosophy & Practice **7**.

Okamoto, K., H. Asanuma, et al. (2014). A graph based data mining method for collaborative learning space in learning commons. World Automation Congress (WAC), 2014.

Okojie, V. (2010). "Innovative financing for university libraries in sub-Saharan Africa." Library Management **31**(6): 404-419.

Pallis, G., L. Angelis, et al. (2005). Model-based cluster analysis for Web users sessions. In Proceedings of the 15th international symposium on methodologies for intelligent systems (ISMIS 2005), Saratoga (NY) USA.

Pallis, G., L. Angelis, et al. (2007). "Validation and interpretation of Web users' sessions clusters." Information Processing & Management **43**(5): 1348-1367.

Palmer, B. C. (2012). Web usage mining: Application to an online educational digital library service. Instructional Technology and Learning Sciences. Logan, Utah, Utah State University. **DOCTOR OF PHILOSOPHY:** 263.

Pani, S. (2011). "Web Usage Mining: A survey on pattern extraction from web logs." International Journal of Control Automation and Systems **1**: 10-19.

Papagelis, M., D. Plexousakis, et al. (2005). Alleviating the sparsity problem of collaborative filtering using trust inferences. Proceedings of the Third international conference on Trust Management. Paris, France, Springer-Verlag**:** 224-239.

Papic, A. and M. Primorac (2014). Introducing e-CRM into academic libraries: Exploration of needs and possibilities. Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on.

Park, W.-B. C., Young-Sung; Ko, Hyung-Hwa (2013). "A Study on Hybrid Recommendation System Based on Usage Frequency for Multimedia Contents." Journal of Digital Contents Society **14**(4): 419-428.

Parsons, J., P. Ralph, et al. (2011). Using Viewing Time to Infer User Preference in Recommender Systems AAAI Workshop in Semantic Web Personalization. San Jose, California.

Patel, V. R. and R. G. Mehta (2011). "Impact of Outlier Removal and Normalization Approach in Modified k-Means Clustering Algorithm." IJCSI International Journal of Computer Science and Information Security(IJCSIS) **8**(5): 331-336.

Pazzani, M. J. (1999). "A Framework for Collaborative, Content-Based and Demographic Filtering." Artif. Intell. Rev. **13**(5-6): 393-408.

Pazzani, M. J. and D. Billsus (2007). Content-based recommendation systems. The adaptive web. B. Peter, K. Alfred and N. Wolfgang, Springer-Verlag**:** 325-341.

Pei, J., J. Han, et al. (2000). Mining Access Patterns Efficiently from Web Logs. Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications, Springer-Verlag**:** 396-407.

Peska, L. and P. Vojtas (2013). Negative implicit feedback in e-commerce recommender systems. Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics. Madrid, Spain, ACM**:** 1-4.

Peska, L. and P. Vojtas (2015). Using Implicit Preference Relations to Improve Content Based Recommending. E-Commerce and Web Technologies: 16th International Conference on Electronic Commerce and Web Technologies, EC-Web 2015, Valencia, Spain, September 2015, Revised Selected Papers. H. Stuckenschmidt and D. Jannach. Cham, Springer International Publishing**:** 3-16.

Pessiot, J.-F., T.-V. Truong, et al. (2007). Learning to Rank for Collaborative Filtering. Proceedings of the Ninth International Conference on Enterprise Information Systems(ICEIS 2007), Funchal, Madeira, Portugal.

Pinto, M. A. G., R. Tanscheit, et al. (2012). Hybrid recommendation system based on collaborative filtering and fuzzy numbers. 2012 IEEE International Conference on Fuzzy Systems.

Pohl, S. (2006). Using Access Data for Paper Recommendations on ArXiv.org, Technische Universit. **Master's thesis**.

Pohl, S., F. Radlinski, et al. (2007). Recommending related papers based on digital library access records. Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries. Vancouver, BC, Canada, ACM**:** 417-418.

Popescul, A., L. H. Ungar, et al. (2001). Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc.**:** 437-444.

Porcel, C., J. M. M. d. Castillo, et al. (2010). "An improved recommender system to avoid the persistent information overload in a university digital library." Control and Cybernetics **39**(4): 900-923.

Porcel, C. and E. Herrera-Viedma (2009). A Fuzzy Linguistic Recommender System to Disseminate the Own Academic Resources in Universities. Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on.

Porcel, C. and E. Herrera-Viedma (2010). "Dealing with incomplete information in a fuzzy linguistic recommender system to disseminate information in university digital libraries." Knowledge-Based Systems **23**(1): 32-39.

Porcel, C., J. M. Moreno, et al. (2009). "A multi-disciplinar recommender system to advice research resources in University Digital Libraries." Expert Systems with Applications **36**(10): 12520-12528.

Quiroga, L. M. and J. Mostafa (2002). "An experiment in building profiles in information filtering: the role of context of user relevance feedback." Information Processing & Management **38**(5): 671-694.

Radlinski, F. and T. Joachims (2005). Query chains: learning to rank from implicit feedback. Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. Chicago, Illinois, USA, ACM**:** 239-248.

Rahman, H. (2008). Data Mining Applications for Empowering Knowledge Societies, IGI Publishing.

Rajagopal, S. and A. Kwan (2012). Book Recommendation System using Data Mining for the University of Hong Kong Libraries. CITERS Conference. Hong Kong, Centre for Information Technology in Education, Faculty of Education, University of Hong Kong.

Ramakrishna, M. T., L. K. Gowdar, et al. (2010). Web Mining: Key Accomplishments, Applications and Future Directions. Data Storage and Data Engineering (DSDE), 2010 International Conference on.

Razilan, A.K., et al. (2009). Academic Digital Library's Evaluation Criteria: User-Centered Approach. Proceeding of ICDL09, Paris.

Recker, M. M., A. E. Walker, et al. (2007) A Study of Teachers' Use of Online Learning Resrouces to Design Classroom Activities. New Review of Hypermedia and Multimedia **13**, 117-134 DOI: 10.1080/13614560701709846

Renda, M. E. and U. Straccia (2005). "A personalized collaborative digital library environment: a model and an application." Inf. Process. Manage. **41**(1): 5-21.

Rendle, S., C. Freudenthaler, et al. (2009). BPR: Bayesian personalized ranking from implicit feedback. Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. Montreal, Quebec, Canada, AUAI Press**:** 452-461.

Ricci, F., L. Rokach, et al. (2011). Introduction to Recommender Systems Handbook. Recommender Systems Handbook, Springer**:** 1-35.

Roh, T. H., K. J. Oh, et al. (2003). "The collaborative filtering recommendation based on SOM cluster-indexing CBR." Expert Systems with Applications **25**(3): 413-423.

Romero, C. and S. Ventura (2007). "Educational data mining: A survey from 1995 to 2005." Expert Systems with Applications **33**(1): 135-146.

Romero, C. and S. Ventura (2007). "Educational data mining: A survey from 1995 to 2005." Expert Syst. Appl. **33**(1): 135-146.

Romero, C., S. Ventura, et al. (2007). Personalized Links Recommendation Based on Data Mining in Adaptive Educational Hypermedia Systems. Creating New Learning Experiences on a Global Scale. E. Duval, R. Klamma and M. Wolpers, Springer Berlin Heidelberg. **4753:** 292-306.

Ross, L. and P. Sennyey (2008). "The Library is, Dead, L'ong Live the Library! The Practice of, Academic Librarianship and the Digital evolution." The Journal of Academic Librarianship **24**(2): 145-152.

Runhua, W., T. Yi, et al. (2011). K-means clustering algorithm application in university libraries. Cognitive Informatics & Cognitive Computing (ICCI*CC), 2011 10th IEEE International Conference on.

Sahoo, N., P. V. Singh, et al. (2012). "A hidden Markov model for collaborative filtering." MIS Q. **36**(4): 1329-1356.

Samizadeh, R. and B. Ghelichkhani (2010). "Use of Semantic Similarity and Web Usage Mining to Alleviate the Drawbacks of User-Based Collaborative Filtering Recommender Systems." International Journal of Industiral Engineering & Producion Research **21**(3): 137-146.

Sarwar, B., G. Karypis, et al. (2000). Analysis of recommendation algorithms for e-commerce. Proceedings of the 2nd ACM conference on Electronic commerce. Minneapolis, Minnesota, USA, ACM**:** 158-167.

Sarwar, B., G. Karypis, et al. (2001). Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th international conference on World Wide Web. Hong Kong, Hong Kong, ACM**:** 285-295.

Sarwar, B. M., J. Konstan, et al. (2002). Recommender Systems for Large-scale E-Commerce: Scalable Neighborhood Formation Using Clustering. Conference on Computer and Information Technology.

Savolainen, R. (2007). "Filtering and withdrawing: strategies for coping with information overload in everyday contexts." J. Inf. Sci. **33**(5): 611-621.

Schafer, J. B., D. Frankowski, et al. (2007). Collaborative filtering recommender systems. The adaptive web. B. Peter, K. Alfred and N. Wolfgang, Springer-Verlag**:** 291-324.

Schein, A. I., A. Popescul, et al. (2002). Methods and metrics for cold-start recommendations. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. Tampere, Finland, ACM**:** 253-260.

Seo, Y.-W. and B.-T. Zhang (2000). Learning user's preferences by analyzing Web-browsing behaviors. Proceedings of the fourth international conference on Autonomous agents. Barcelona, Spain, ACM**:** 381-387.

Shani, G. and A. Gunawardana (2009). Evaluating Recommender Systems.

Shardanand, U. and P. Maes (1995). Social information filtering: algorithms for automating "word of mouth". Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Denver, Colorado, USA, ACM Press/Addison-Wesley Publishing Co.**:** 210-217.

Sharifabadi, S. R. (2006). "How digital libraries can support e-learning." The Electronic Library **24**(3): 389-401.

Shenoy, P., M. Jain1, et al. (2013). "Web Usage Mining Using Pearson's Correlation Coefficient." International Journal of Engineering Research and Applications (IJERA) **3**(2): 676-679.

Si, S., A. D. Sarma, et al. (2014). Beyond modeling private actions: predicting social shares. Proceedings of the 23rd International Conference on World Wide Web. Seoul, Korea, International World Wide Web Conferences Steering Committee**:** 377-378.

Singh, B. and H. K. Singh (2010). Web Data Mining research: A survey. Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International Conference on.

Sitanggang, I. S., N. A. Husin, et al. (2010). Sequential pattern mining on library transaction data. Information Technology (ITSim), 2010 International Symposium in.

Smeaton, A. F. and J. Callan (2005). "Personalisation and recommender systems in digital libraries." International Journal on Digital Libraries **5**(4): 299-308.

Smith, M. Q. (2005). The impact of information and communications technology change on the management and operations of academic libraries. Department of Library and Information Science, University of the Western Cape. **Magister Bibliothecologiae - MBibl**.

Smyth, B., J. Freyne, et al. (2004). I-SPY — Anonymous, Community-Based Personalization by Collaborative Meta-Search. Research and Development in Intelligent Systems XX. F. Coenen, A. Preece and A. Macintosh, Springer London**:** 367-380.

Soboroff, I. and C. Nicholas (2000). Collaborative filtering and the generalized vector space model (poster session). Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. Athens, Greece, ACM**:** 351-353.

Soria, K. M., J. Fransen, et al. (2013). Library Use and Undergraduate Student Outcomes: New Evidence for Students' Retention and Academic Success.

Speretta, M. and S. Gauch (2005). Personalized Search Based on User Search Histories. Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence, IEEE Computer Society**:** 622-628.

Spink, A., T. D. Wilson, et al. (2002). "Information-seeking and mediated searching. Part 1. Theoretical framework and research design." Journal of the American Society for Information Science and Technology **53**(9): 695-703.

Srivastava, J., R. Cooley, et al. (2000). "Web usage mining: discovery and applications of usage patterns from Web data." SIGKDD Explor. Newsl. **1**(2): 12-23.

Stojanovski, J. and A. Papic (2012). Quantitative indicators of academic libraries' involvement in educational process. Information Technology Interfaces (ITI), Proceedings of the ITI 2012 34th International Conference on.

Su, X. and T. M. Khoshgoftaar (2009). "A survey of collaborative filtering techniques." Adv. in Artif. Intell. **2009**: 2-2.

Suguna, R. and D. Sharmila (2013). "An Efficient Web Recommendation System using Collaborative Filtering and Pattern Discovery Algorithms." International Journal of Computer Applications **70** (3): 37-44.

Sun, X., F. Kong, et al. (2005). Using latent class models for neighbors selection in collaborative filtering. Proceedings of the First international conference on Advanced Data Mining and Applications. Wuhan, China, Springer-Verlag**:** 149-156.

Suneetha, K. R. and R. Krishnamoorthi (2009). Identifying User Behavior by Analyzing Web Server Access Log File.

Suresh, R. M. (2007). A study on the ontology based Web mining for digital library. Information and Communication Technology in Electrical Sciences (ICTES 2007), 2007. ICTES. IET-UK International Conference on.

Sushmitha, S., N. Annushya, et al. (2015). "QOS AWARE SPARSE DATA PERSONALIZED RECOMMENDATION SYSTEM." International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) **13**(1).

Symeonidis, P., A. Nanopoulos, et al. (2008). "Providing Justifications in Recommender Systems." IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans **38**(6): 1262-1272.

Symeonidis, P., A. Nanopoulos, et al. (2008). "Collaborative recommender systems: Combining effectiveness and efficiency." Expert Systems with Applications **34**(4): 2995-3013.

Tejeda-Lorente, A., J. Bernabé-Moreno, et al. (2014). "Integrating Quality Criteria in a Fuzzy Linguistic Recommender System for Digital Libraries." Procedia Computer Science **31**(0): 1036-1043.

Tejeda-Lorente, A., C. Porcel, et al. (2011). Using memory to reduce the information overload in a university digital library. Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on.

Tejeda-Lorente, A., C. Porcel, et al. (2014). "A quality based recommender system to disseminate information in a university digital library." Inf. Sci. **261**: 52-69.

Tingting, Z. and Z. Lili (2011). Application of data mining in the analysis of needs of university library users. Computer Science & Education (ICCSE), 2011 6th International Conference on.

Tsai, C. S. and M. Y. Chen (2008). "Using adaptive resonance theory and data-mining techniques for materials recommendation based on the e-library environment." The Electronic Library **26**(3): 287-302.

Tung-Shou, C., L. Ming-Horng, et al. (2004). Enhancing library resources usage efficiency by data mining. Networking, Sensing and Control, 2004 IEEE International Conference on.

Umamaheswari, S. and S. K. Srivatsa (2014). "Algorithm for Tracing Visitors' On-Line Behaviors for Effective Web Usage Mining." International Journal of Computer Applications **87**(3).

Upadhyay, N. (2015). Trends that will affect technology and resource decision in academic libraries in near future. Emerging Trends and Technologies in Libraries and Information Services (ETTLIS), 2015 4th International Symposium on.

Uppal, V. and G. Chindwani (2013). "An Empirical Study of Application of Data Mining Techniques in Library System." International Journal of Computer Applications **74**(11): 42-46.

V.Chitraa and A. S. Davamani (2010). "A Survey on Preprocessing Methods for Web Usage Data." International Journal of Computer Science and Information Security(IJCSIS) **7**(3).

Varnagar, C. R., N. N. Madhak, et al. (2013). Web usage mining: A review on process, methods and techniques. Information Communication and Embedded Systems (ICICES), 2013 International Conference on.

Vellino, A. (2010). "A comparison between usage-based and citation-based methods for recommending scholarly research articles." Proceedings of the American Society for Information Science and Technology **47**(1): 1-2.

Virmani, D., Shweta Taneja, et al. (2015). "Normalization based K means Clustering Algorithm." International Journal of Advanced Engineering Research and Science **2** (2).

Volkovs, M. and G. W. Yu (2015). Effective Latent Models for Binary Feedback in Recommender Systems. Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Santiago, Chile, ACM**:** 313-322.

Wahab, M. H. A., M. N. H. Mohd, et al. (2008). "Data Pre-processing on Web Server Logs for Generalized Association Rules Mining Algorithm." Proceedings of World Academy of Science: Engineering & Technolog **40**.

Wan-Shiou, Y., D. Jia-Ben, et al. (2006). Mining Social Networks for Targeted Advertising. System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on.

Wang, B., R. Xu, et al. (2005). Study on Applications of Web Mining to Digital Library. Artificial Intelligence Applications and Innovations. D. Li and B. Wang, Springer US. **187:** 777-787.

Wang, T. and Y. Ren (2009). "Research on personalized recommendation based on web usage mining using collaborative filtering technique." WSEAS Trans. Info. Sci. and App. **6**(1): 62-72.

Wang, Z., X. Yu, et al. (2014). "An improved collaborative movie recommendation system using computational intelligence." Journal of Visual Languages & Computing **25**(6): 667-675.

WeiJi, S. Liu1, et al. (2016). Research of Intelligent recommendation system based on the user and association rules mining for books. 5th International Conference on Computer Sciences and Automation Engineering (ICCSAE 2015), Atlantis Press**:** 294 - 299.

White, R. W., J. M. Jose, et al. (2006). "An implicit feedback approach for interactive information retrieval." Inf. Process. Manage. **42**(1): 166-190.

Winoto, P., T. Y. Tang, et al. (2012). "Contexts in a paper recommendation system with collaborative filtering." International Review of Research in Open and Distance Learning **13**(5): 56-75

Witten, I. H., D. Bainbridge, et al. (2010). How to Build a Digital Library. How to Build a Digital Library (Second Edition). I. H. Witten, D. Bainbridge and D. M. Nichols. Boston, Morgan Kaufmann.

Wu, C. H. (2003). "Data mining applied to material acquisition budget allocation for libraries: design and development." Expert Systems with Applications **25**(3): 401-411.

Wu, Z., B. H. Tan, et al. (2015). Neural Modeling of Buying Behaviour for E-Commerce from Clicking Patterns. Proceedings of the 2015 International ACM Recommender Systems Challenge. Vienna, Austria, ACM**:** 1-4.

Xiaojian, L. and W. Yuchun (2012). Borrowing Data Mining Based on Association Rules. Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on.

Xingyuan, L. (2011). Collaborative filtering recommendation algorithm based on cluster. Computer Science and Network Technology (ICCSNT), 2011 International Conference on.

Xu, B. and M. M. Recker (2011). "Understanding Teacher Users of a Digital Library Service: A Clustering Approach." Journal of Educational Data Mining **3**(1).

Xu, G. (2008). Web mining techniques for recommendation and personalization. The School of Computer Science & Mathematics, Faculty of Health, Engineering & Science. Victoria University, Australia Victoria University. **PhD**.

Xu, G., Y. Zhang, et al. (2005). A Web Recommendation Technique Based on Probabilistic Latent Semantic Analysis. Web Information Systems Engineering – WISE 2005. A. H. Ngu, M. Kitsuregawa, E. Neuhold, J.-Y. Chung and Q. Sheng, Springer Berlin Heidelberg. **3806:** 15-28.

Xue, G.-R., C. Lin, et al. (2005). Scalable collaborative filtering using cluster-based smoothing. Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. Salvador, Brazil, ACM**:** 114-121.

Xuesong, Z. and J. Kaifan (2013). Tourism e-commerce recommender system based on web data mining. Computer Science & Education (ICCSE), 2013 8th International Conference on.

Yang, C., B. Wei, et al. (2009). CARES: a ranking-oriented CADAL recommender system. Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries. Austin, TX, USA, ACM**:** 203-212.

Yang, D., T. Chen, et al. (2012). Local implicit feedback mining for music recommendation. Proceedings of the sixth ACM conference on Recommender systems. Dublin, Ireland, ACM**:** 91-98.

Yang, Y., X. Wang, et al. (2014). "A multi-dimensional image quality prediction model for user-generated images in social networks." Information Sciences **281**: 601-610.

Yuanyuan, W., S. C. F. Chan, et al. (2012). Applicability of Demographic Recommender System to Tourist Attractions: A Case Study on Trip Advisor. Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012 IEEE/WIC/ACM International Conferences on.

Yung, C. (2015). Mining Massive Web Log Data of an Official Tourism Web Site as a Step towards Big Data Analysis in Tourism. Proceedings of the ASE BigData & SocialInformatics 2015. Kaohsiung, Taiwan, ACM**:** 1-4.

Zanker, M., M. Fuchs, et al. (2008). Evaluating Recommender Systems in Tourism — A Case Study from Austria. Information and Communication Technologies in Tourism 2008. P. O'Connor, W. Höpken and U. Gretzel, Springer Vienna**:** 24-34.

Zhang, M. (2011). "Application of Data Mining Technology in Digital Library " JOURNAL OF COMPUTERS **6(4)**.

Zhao, J. (2011). Web mining application in university library personalized search engine. 2011 International Conference of Soft Computing and Pattern Recognition (SoCPaR).

Zhao, J. and P. O. de Pablos (2011). "Regional knowledge management: the perspective of management theory." Behaviour & Information Technology **30**(1): 39-49.

Zhao, J., P. O. de Pablos, et al. (2012). "Enterprise knowledge management model based on China's practice and case study." Computers in Human Behavior **28**(2): 324-330.

Zhao, Y., Z. Niu, et al. (2014). "Research on Data Mining Technologies for Complicated Attributes Relationship in Digital Library Collections." Applied Mathematics & Information Sciences **8(3)**: 1173-1178.

Zhou, P. and Z. Le (2007). A Framework for Web Usage Mining in Electronic Government. Integration and Innovation Orient to E-Society Volume 2. W. Wang, Y. Li, Z. Duanet al, Springer US. **252:** 487-496.

Zhu, J. and Q. Xu (2011). The Research on Exploring E-Commerce Model for Academic Library in China. Management and Service Science (MASS), 2011 International Conference on.

Zhu, Z. and J.-y. Wang (2007). Book Recommendation Service by Improved Association Rule Mining Algorithm. Machine Learning and Cybernetics, 2007 International Conference on.

Zorrilla, M., D. García, et al. (2010). A decision support system to improve e-learning environments. Proceedings of the 2010 EDBT/ICDT Workshops. Lausanne, Switzerland, ACM**:** 1-8.