# A Data Mining Framework for Building A Web-Page Recommender System

Choochart Haruechaiyasak
Information Research and Development Division (RDI)
National Electronics and Computer Technology Center (NECTEC)
Thailand Science Park, Klong Luang, Pathumthani 12120, Thailand
choochart.haruechaiyasak@nectec.or.th

Mei-Ling Shyu
Department of Electrical and Computer Engineering, University of Miami
Coral Gables, FL 33124, USA
shyu@miami.edu

Shu-Ching Chen
Distributed Multimedia Information System Laboratory, School of Computer Science
Florida International University, Miami, FL 33199, USA
chens@cs.fiu.edu

## Abstract

*In this paper, we propose a new framework based on data mining algorithms for building a Web-page recommender system. A recommender system is an intermediary program (or an agent) with a user interface that automatically and intelligently generates a list of information which suits an individual's needs. Two information filtering methods for providing the recommended information are considered: (1) by analyzing the information content, i.e., content-based filtering, and (2) by referencing other user access behaviors, i.e., collaborative filtering. By using the data mining algorithms, the information filtering processes can be performed prior to the actual recommending process. As a result, the system response time could be improved and thus, making the framework scalable.*

## 1. Introduction

To alleviate the *information overload* [7] problem (especially on the World Wide Web), traditional Information Retrieval (IR) techniques have been employed to assist the users in finding their information. One of the most widely applied IR techniques for assisting the users in finding their information is the keyword-based search, as adopted by many Web search engines. However, without prior knowledge of the retrieval process, or the keywords which accurately depict the search topic, discovering the desired information can be a tedious and formidable task. In addition, the traditional IR approaches have no way to customize the results, according to the users' preferences.

Recently, the research within the IR community has considered an alternative approach of retrieving the information based on the recommendation of the users in the system. Many recommender systems have been designed and implemented for various types of items including newspaper, research papers, emails, Usenet news, books, movies, music, restaurants, Web pages, and e-commerce products [8, 9, 10, 12]. To construct a recommender system, two information filtering methods for providing the recommended information are considered: (1) by analyzing the information content, i.e., content-based filtering, and (2) by referencing other users' access behaviors, i.e., collaborative filtering. However, most of the existing approaches consider only one of the filtering techniques in their recommending processes. In order to recommend the information to a particular user, the original collaborative filtering technique relies on the Nearest Neighbor (NN) clustering algorithm to compare and search for other users' ratings which are closely resemble to the user's profile [9, 13]. The Nearest Neighbor clustering algorithm, however, suffers from the scalability problem, i.e., the execution time grows linearly with the number of the users in the system. This collaborative filtering approach also suffers from the *cold-start* problem. Another alternative technique called content-based filtering has additionally been considered in the recommender

system [1, 2]. The content-based filtering is based on the analysis of the information contents, e.g., by applying a classification method on textual content.

In this paper, a new framework is proposed that combines both content-based and collaborative filtering and utilizes data mining techniques. Our proposed content-based filtering integrates both textual analysis and the user personalization during the recommendation process; while the collaborative filtering is based on the method of mining user access patterns, which is performed by applying the association rule mining on the user access sequences with the traversal constraints. Applying the association rule mining for mining user access patterns allows the prediction of the Web pages to include additional non-consecutive Web pages, and thus enhances the prediction performance in terms of precision and recall. Also, the experimental results showed that our method has a better potential of reducing the user access time on the Web site compared to the existing Markov model approach [5, 6]. Applying data mining algorithms, e.g., association rule mining, for the information filtering techniques provides efficiency, since the recommended list of information can be generated prior to the recommendation process. Therefore, the response time of the recommender system could be improved. In addition, since data mining algorithms with the use of data reduction and selection techniques are typically capable of handling an enormous amount of data, the proposed framework is scalable for larger Web sites and domains.

This proposed framework is then applied to construct a system prototype for recommending the University of Miami (UM)'s Web pages, which automatically generates a recommended list of Web pages based on an individual's preferences. The framework is based on the concept of information filtering which refers to the process of automatically screening the information based on the user's specification. Our Web-page recommender system enhances the functionality of a typical query-based information retrieval system (e.g., search engine) by applying the information filtering techniques to automatically generate a recommended list of Web pages which personalize each individual's preferences. User personalization is an important component which distinguishes a recommender system from a typical search engine. By allowing the system to monitor each user profile, a system can learn from user's past preferences and thus, yield more effective retrieval results. To provide the efficiency of data and information access and retrieval, a database for the recommender system is designed and implemented. The database is designed to store all the related data components such as textual content, link structure, recommended list of Web pages, and user profiles. To provide a Web-based user interface, the database is stored and linked to a Web server. The Web server is implemented on the HTTP (HyperText Transfer Protocol) specification.

The remainder of this paper is organized as follows. In the next section, the proposed data mining framework for building a Web-page recommender system is presented. In Section 3, a system prototype for recommending Web pages is given along with the detailed implementation. The paper concludes in Section 4.

## 2. The proposed recommender system framework

Data mining or knowledge discovery in databases (KDD) has emerged recently as an active research area for extracting implicit, previously unknown, and potentially useful information from large databases [3, 4]. In this paper, we apply data mining techniques into the IR context, specifically as the information filtering tools for the proposed framework. Two types of information filtering can be accomplished by using data mining: content-based filtering and collaborative filtering. In the proposed framework, the association rule mining technique is applied as the content-based filtering where the data set is the keyword matrix (size: Web documents $\times$ keywords). Collaborative or social-based filtering retrieves the information for a particular user by referring to other user evaluations on the information content. The method of mining user access patterns based on the association rule mining is applied as the collaborative filtering technique. The overall process for designing and implementing a recommender system involves the following five steps.

- Data Collection: This initial step involves the collection of data sets for executing the data mining algorithms. Three data components are considered: (1) textual content (i.e., index terms or keywords), (2) link structure (embedded hyperlinks within Web pages), and (3) user log records.

- Data Preprocessing: This step is required to clean and transform the collected data sets into the formats which are suitable for the data mining algorithms. This step includes the data reduction and selection techniques to improve the efficiency of the data mining algorithms.

- Information Filtering via Data Mining: This step is the core process of the recommender system framework, where the data sets are analyzed and the data mining algorithms are applied as the information filtering tools to generate and discover any useful and interesting recommended outputs.

- Database Design and Implementation: To improve the efficiency of data and information access and retrieval, the database for the recommender system is designed and implemented for all related data sets including the

textual content, link structure, and the recommended lists of Web pages.

- User Interface Design and Implementation: The user interface acts as an intermediary between the users and the recommender system. This step involves the design and implementation of a Web (i.e., HTTP) server which receives the users' requests via WWW, processes the requests by accessing the database, and responds by returning the results to the users. The user interface provides a recommendation function with the user personalization technique by requiring each user to log into the system in order to keep track of the preferences.

## 3. A system prototype for recommending Web pages

The Web-page recommender system prototype focuses on the UM's Web site as the main scope. Similar to many other educational institutions, UM provides the information including other Web services through its own Web site. The main groups of people who access the UM's Web site include students, faculty, staff, and general public who are interested in finding the information about the university. The Web site has the main page which can be accessed via *www.miami.edu*. Currently, the information can be accessed by either (1) using the query-based search or (2) browsing through the hyperlinks within the Web pages. The recommender system is designed to enhance these two retrieval methods of the Web site by generating the list of Web pages which match the user's interests.

### 3.1 Data collection and preprocessing

Figure 1 illustrates the data collection and preprocessing steps. The UM's Web server system contains two types of databases: (1) the content database for providing the information content such as Web pages with the images and other types of data, and (2) the server log database for recording the HTTP transactions (i.e., log records).

To collect the raw data sets from the Web site, a *crawling* and *parsing* program was implemented using the *JAVA* programming language. The number of unique Web pages collected from the Web site was *21,446*. Two different data sets were collected from the crawling and parsing process: the textual and hyperlink contents. The textual content was cleaned through the stemming and stopword removal process to form the words and to remove the non-discriminant words. By applying the *document-frequency* keyword selection technique [14], a set of selected keywords are extracted. The number of the keywords is limited to *10,000*. Therefore, the keyword matrix of *21,446* documents by

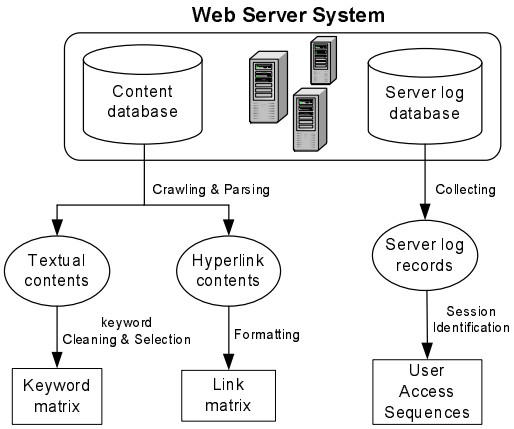


**Figure 1. Data collection and preprocessing**

*10,000* keywords can be constructed. The link structure of the Web site can be constructed by extracting the embedded hyperlinks within each Web page. In the WWW environment, each hyperlink refers directly to a physical Web page. Therefore, the hyperlinks and the actual Web pages are used interchangeably. In this study, the link adjacency matrix of *21,446* documents by *21,446* hyperlinks can be constructed. Since both keyword and link matrices are sparse, therefore, the alternative adjacency-list data structure is used to reduce the space complexity.

Another data set which needs to be collected is the user log records. Under HTTP, each request to the Web site is recorded as a transaction on the Web server. UM has two Web servers. Due to the load-balancing configuration, the HTTP traffic volumes of both servers are closely similar. From the traffic volumes, the first observation is that both servers yield a similar amount of daily traffic. Another observation is the patterns in a weekly traffic cycle, i.e., the traffic volume is higher during the weekdays and relatively lower during the weekends. Based on this observation, to reduce the execution time, a traffic volume of one-week period is used to represent the input to the Web mining process, which the log records from both servers are combined into a single file using the *Merge Sort* algorithm [11] based on the timestamp ordering. Since the scope of this recommender system is only for the Web pages or the *HyperText Markup Language* (*HTML*) documents, other irrelevant items such as images can be eliminated by removing the log records containing the filename suffixes such as *gif* and *jpg*. Next, user access sequences are constructed based on the session identification. A user access sequence, also referred to as a browsing sequence or a traversal path, is an ordered list of Web pages accessed by a user during one session. To identify a user session, the IP addresses of the requesting machines are used to distinguish individual users. Also, the timeout period is included as a heuristic to

identify different users who use the same computers with the same IP address. Timeout refers to the total time period an average user would spend on accessing the Web site. Using the above constraints, approximately *50,000* user access sequences are constructed from the log records of the one-week period.

### 3.2 Information filtering via data mining algorithms

Figure 2 illustrates the information filtering process by applying the data mining algorithms. Content-based filtering is achieved by applying the association rule mining technique on the keyword matrix. The keyword matrix of *21,446* documents by *10,000* keywords is first transposed so that each keyword is used to represent a record for the association rule mining algorithm. The results are a set of rules in the form of "IF (preconditioned set of Web pages) THEN (post-conditioned Web page)." In order to simplify the model, only the single-consequent rules are considered. If the same precondition occurs in more than one rule, the post-conditioned Web pages are ranked based on the confidence values of the rules. The content-based filtering rules are such that the preconditioned Web pages imply the post-conditioned Web page based on the similarity in the keywords (i.e., textual content).

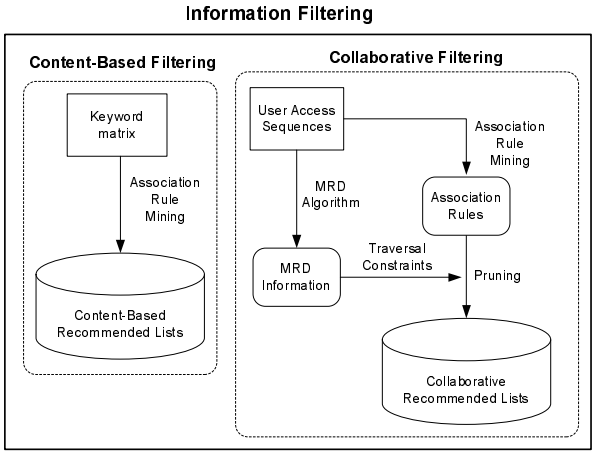


**Figure 2. Information filtering by applying data mining algorithms**

Also shown in Figure 2 is the collaborative or social-based filtering process. This process applies the association rule mining technique on the user access sequences in order to generate a set of rules. Then by using the user access sequences, the traversal constraint called the Minimum Reaching Distance (MRD) information is constructed based on the shortest-path problem in graph theory. The MRD information captures the actual user access behaviors on the Web site. The MRD information is then used to prune the rules obtained from the association rule mining process in order to increase the prediction accuracy and reduce the space complexity. The experiments in [5, 6] showed that by using the association rule mining with the MRD information as the traversal constraint, the performance in terms of precision and recall improved over the Markov model method in mining user access patterns.

### 3.3 Database design and implementation with a Web-based user interface

In our proposed framework, a database using the Relational Database Management System (RDBMS) is designed and implemented. This database stores the URLs (i.e., Web pages), keywords for the Web pages, the recommended set of rules from content-based filtering, the recommended set of rules from social-based filtering, user login information, and user profiles. The current version of the recommender system prototype uses MySQL [17] as the choice for database implementation. MySQL provides a multi-threaded, multi-user, and robust SQL (Structured Query Language) database management system, which is suitable for the application of recommender systems.

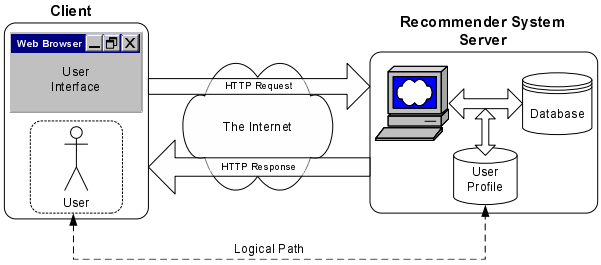


**Figure 3. The interaction of the user and the recommender system via HTTP.**

To provide a Web-based interface, the database is stored and linked to a Web server. The Web server is implemented on the HTTP specification and has the following functions: listening for HTTP requests on a network, receiving HTTP requests made by user agents (usually Web browsers), serving the requests (accessing the database), and returning HTTP response that contains the requested resources. Figure 3 illustrates the interaction between the user and the recommender system under the HTTP specification. The user accesses the recommender system by using a Web browser. The communication between the user and the system is carried out on the Internet via the HTTP request and response functions. The recommender system server provides the database which contains the processed

information such as the recommended lists. The user profile resides on the server side to keep track of the user's preferences. A logical path exists between the user and his/her profile.

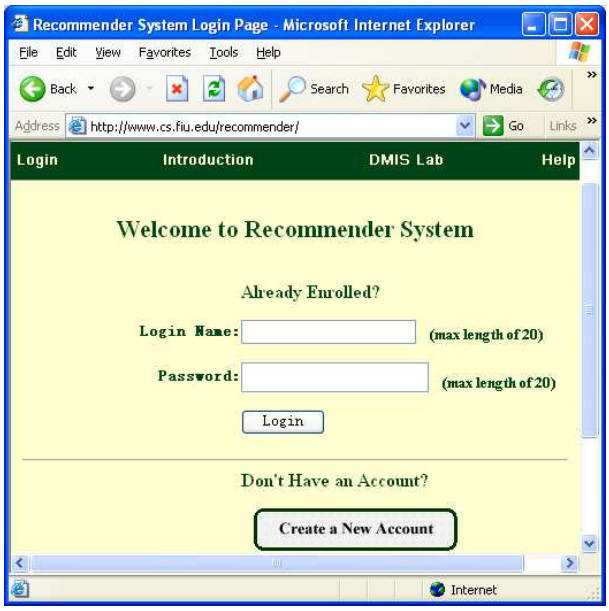


**Figure 4. A snapshot of the recommender system interface: user login page.**

The Web server for our system prototype is implemented by using the Apache HTTP Server software package [15]. The main advantage of using the Apache HTTP server is its open-source feature which supports modern operating systems including *UNIX* and *Microsoft Windows*. The user interface is implemented by using both HTML and PHP (PHP: Hypertext Preprocessor) [16]. While HTML provides a simple and easy way to present the information to the users, PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. PHP is used along with the Apache HTTP Server and MySQL to allow the HTTP requests to be processed, the database to be accessed, and the HTTP responses to be sent to the users. Figure 4 shows a snapshot of the user login page of the recommender system viewed by using a Web Browser.

To obtain the recommended list of Web pages, the ratio percentage between the content and social filtering portions can be specified. For example, by specifying 10% from the *radio* button list, 10% of the recommended Web pages are generated from the content-based filtering list and 90% from the social-based filtering list. Figure 5 shows a snapshot of the recommender system interface by specifying 10% on the recommendation selection. The recommended list of Web pages which match the user's profile is

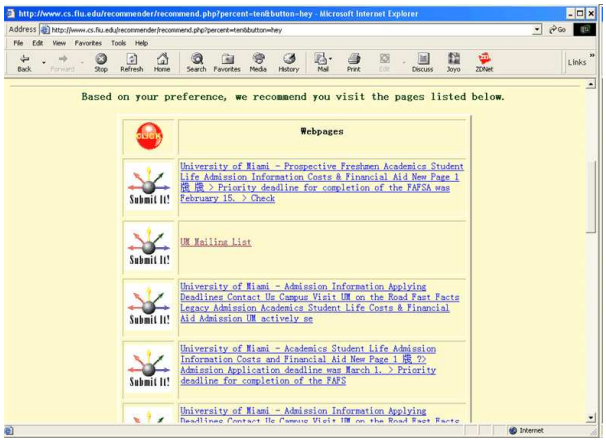


**Figure 5. A snapshot of the recommender system interface: by using recommendation.**

shown on the bottom section of the page. To view the Web page, the user clicks on the URL name, and the actual Web page will be opened in a new window. If the Web page has the information which matches the user's interest, the user must click on the button in front of the URL name in order to record the preference into his/her profile. By providing the preference of Web pages, the system can recommend Web pages which match the user's interest. Our proposed recommender system also provides other system functions including (1) "view your past ten pages" and (2) "help" page. The first function shows the past ten Web pages as recorded in the user profile, while the second one provides the general information of the recommender system as well as the instruction of how to use the system.

We conducted a qualitative performance evaluation based on some user surveys. A group of 20 users were independently asked to evaluate the system by giving a score in the scale of 1 (strongly disagree) to 5 (strongly agree) to each of the eleven questions. These questions are related to the effectiveness of different retrieval methods and other features provided by the system. In summary, most users agree that by using the recommending feature, the system could yield higher satisfactory results than by using the keyword-based search alone (the averaged score of 4.1 versus 3.3). Another important survey result is that most users prefer the system's ability to adjust the filtering scale between content-based and social-based filtering approaches (with the averaged score of 4.3).

## 4. Conclusions

In this paper, a new framework based on data mining techniques is proposed to improve the efficiency and effectiveness of the traditional information retrieval (IR) process.

One of the major obstacles to the traditional IR approaches such as the query-based search, is the lack of effective interaction between the users and the system, which results in inferior retrieved information. The proposed framework is designed to enhance this interaction by analyzing user access behaviors on the system. In addition to the content analysis (i.e., content-based filtering), information is also retrieved according to each individual's preferences (i.e., user personalization) and by recommendation from other users (i.e., collaborative filtering). To demonstrate the potential usage of the framework, a system prototype for recommending Web pages on the UM's Web site is designed and implemented as a form of a Web-site navigational assistant. As opposed to the search engine, in which the users need to formulate a query composing of keywords, the recommender system automatically retrieves the information by analyzing both content and user access behavior. Thus, additional information, which is not directly indexed by the querying keywords but relevant to them, could also be presented to the users. Based on the performance evaluation through the conducted user surveys, the system's recommending feature could yield more satisfactory results than by using the traditional keyword-based search. In addition, the filtering scale-adjusting feature is also found to be useful to most of the users.

## Acknowledgment

For Mei-Ling Shyu, this research was supported in part by NSF ITR (Medium) IIS-0325260. For Shu-Ching Chen, this research was supported in part by NSF EIA-0220562 and NSF HRD-0317692.

## References

[1] M. Balabanovic and Y. Shoham, "Fab: Content-based, collaborative recommendation", *Communications of the ACM*, 40(3):66–72, 1997.

[2] C. Basu, H. Hirsh, and V. Cohen, "Recommendation as classification: Using social and content-based information in recommendation," *Proc. of the Fifteenth National Conf. on Artificial Intelligence*, pp. 714–720, 1998.

[3] M. Chen, J. Han, and P. Yu, "Data mining: an overview from database perspective," *IEEE Trans. on Knowledge and Data Engineering*, 8(6):866–883, Dec. 1996.

[4] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, eds., *Advances in Knowledge Discovery and Data Mining*, AAAI Press, 1996.

[5] C. Haruechaiyasak, M.-L. Shyu, and S.-C. Chen, "A Web-page recommender system via a data mining framework and the semantic Web concept," accepted for publication, *International Journal of Computer Applications in Technology*, Special Issue on Data Mining Applications, 2004.

[6] C. Haruechaiyasak, "A data mining and semantic Web framework for building a Web-based recommender system," Ph.D. dissertation, University of Miami, June 2003.

[7] P. Maes, "Agents that reduce work and information overload," *Communications of the ACM*, 37(7):30–40, 1994.

[8] R. J. Mooney and L. Roy, "Content-based book recommending using learning for text categorization," *Proc. of the Fifth ACM Conf. on Digital Libraries*, pp. 195–204, 2000.

[9] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. T. Riedl, "Item-based collaborative filtering recommendation algorithms," *Proc. of the Tenth Int. WWW Conf.*, pp. 285–295, 2001.

[10] J. B. Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation application," *Data Mining and Knowledge Discovery*, 5(1/2):115–153, 2001.

[11] R. Sedgewick, *Algorithms in C*, Addison-Wesley, 1990.

[12] C. Shahabi, F. Banaei-Kashani, Y.-S. Chen, and D. McLeod, "Yoda: An accurate and scalable Web-based recommendation system," *Proc. of the Sixth Int. Conf. on Cooperative Information Systems*, September 2001.

[13] U. Shardanand and P. Maes, "Social information filtering: Algorithms for automating word of mouth," *Proc. of the Annual ACM SIGCHI on Human Factors in Computing Systems*, pp. 210–217, 1995.

[14] Y. Yang and J. P. Pedersen, "A comparative study on feature selection in text categorization," *Proc. of the Fourteenth Int. Conf. on Machine Learning* pp. 412–420, 1997.

[15] *The Apache HTTP Server Project*, http://httpd.apache.org

[16] *PHP: Hypertext Preprocessor*, http://www.php.net

[17] *MySQL: Open Source Database*, http://www.mysql.com