Extending a Web Browser with Client-Side Mining

Hongjun Lu, Qiong Luo, Yeuk Kiu Shun

Hong Kong University of Science and Technology
Department of Computer Science
Clear Water Bay, Kowloon
Hong Kong, China
{luhj, luo, rayshun}@cs.ust.hk

Abstract. We present WBext (Web Browser extended), a web browser extended with client-side mining capabilities. WBext learns sophisticated user interests and browsing habits by tailoring and integrating data mining techniques including association rules mining, clustering, and text mining, to suit the web browser environment. Upon activation, it automatically expands user searches, re-ranks and returns expanded search results in a separate window, in addition to returning the original search results in the main window. When a user is viewing a page containing a large number of links, WBext is able to recommend a few links from those that are highly relevant to the user, considering both the user's interests and browsing habits. Our initial results show that WBext performs as fast as a common browser and that it greatly improves individual users' search and browsing experience.

1 Introduction

Both individual web sites and common search engines have made significant efforts in organizing their contents and improving search quality in order to ease users' browsing and searching activities. Nevertheless, it is difficult and costly for these sites to tailor their content and service for every single one of the vast web population. Moreover, the interests and browsing habits of individual users are changing over time. Finally, even though some server-side personalization features are available, users usually hesitate to adopt them due to privacy concerns.

Motivated by these problems in server-side personalization, we developed a novel personalized web browser, WBext (Web Browser extended), based on client-side mining. Without the extensions activated, it is just an ordinary web browser (currently using Microsoft Internet Explorer). With the extensions activated, WBext learns user interests and habits from the browser side and provides assistance for the user to locate the desired information at any web sites or search engines.

A screen shot of WBext in use is shown in Figure 1. Click on the menu bars titled "Activity", "Search", "Recommendation", "Setting", and "Log" below the main browser window, the corresponding results from the extensions will be shown in the window at the bottom.

Figure 1. WBext with Internet Explorer

The system has the following novel features:

Automatic, privacy-preserving personalization. WBext uses unsupervised learning; therefore, it does not need the users to predefine their interests or to manage their preferences as previous work based on pre-defined user profiles [5, 13]. Rather, the system continuously monitors user activities, learns user interests and habits, and adapts itself to current user interests. Because it accumulates and updates its knowledge base along with the user's browsing activities, it is able to assist users more effectively over time. In addition, the knowledge discovered is owned by the users for improving their browsing experience, not shared with any web sites for any privacy-violating actions.

Efficient and effective client-side mining. Compared with existing client-side agents supporting personalization, such as the Personal WebWatcher and others [10, 14], WBext tailors and integrates various data mining techniques, including clustering, association mining, and text mining, to discover and to maintain sophisticated user interests in the browser environment. While interests are mainly content-related, WBext further mines user browsing habits, which make link recommendations more focused and truly personalized. Compared with server side usage mining [1, 11, 12], WBext captures user activities more accurately and identifies user interests better. With detailed user activity logs, the system is able to efficiently and reliably resolve sessions and transaction entities, to evaluate importance of different textual contents, as well as to understand users' frequent navigational patterns. This ensures the high quality and completeness of knowledge discovered.

Search Query Expansion. Upon activation, WBext automatically expands a user search to several modified searches and combines and re-ranks the multiple modified search results by utilizing learned knowledge about user interests. This is motivated by a problem with searching using simple keyword queries, which is, users may be interested in only a few links among or other than the highly-ranked ones that a search engine recommends to a general web user base. Search query expansion greatly improves the effectiveness of search attempts for individual users while not requiring their extra effort – they can still input simple keywords as queries. To avoid intervening with the users' normal search environment (for example, their favorite search engine interface), these modified search results are returned in a separate window (as the one at the bottom in Figure 1).

Link Recommendation. Another problem that users often encounter is that an informative web page contains a large number of hyperlinks. In this situation, WBext can recommend a few links ordered by the relevance to a user based on the knowledge learned about her interests and her navigation habits. The recommended links are also shown in the separate window at the bottom in Figure 1.

In the next sections, we present the system components, the learning process, the assistance in user search and browsing, the preliminary experimental results, and our conclusions and future work.

2 WBext: The System

2.1 System Components

Figure 2 shows the system architecture of WBext. It consists of the following sets of main components: WBext Interface, Activity Log, Knowledge Base, WBext Miner, and WBext Agents.

The **WBext Interface** is embedded in the browser and serves as an interface layer between the web browser and other components of the system. It has two major functions: capturing activities that a user performs at the browser, and passing the results of WBext agents to the web browser.

The **Activity Log** stores user activities captured by the WBext Interface. Each entry in the Activity Log is an activity carried out by the user at the web browser. Seven kinds of activities are captured: page visit, search initiation, link following, bookmarking, text selection, page focusing, and new window spawning.

The **Knowledge Base** of WBext has two components, user interests and habits. User interests are frequently searched topics of the user, learned from the user activities. User habits refer to the navigational patterns presented in browsing activities of the user. Both types of knowledge are discovered from the Activity Log.

The **WBext Miner** consists of four modules, Log Processor, Habit Miner, Feature Extractor, and Interest Miner. The Log Processor prepares the activity log for mining. The Feature Extractor and Interest Miner discover knowledge relevant to user inter-

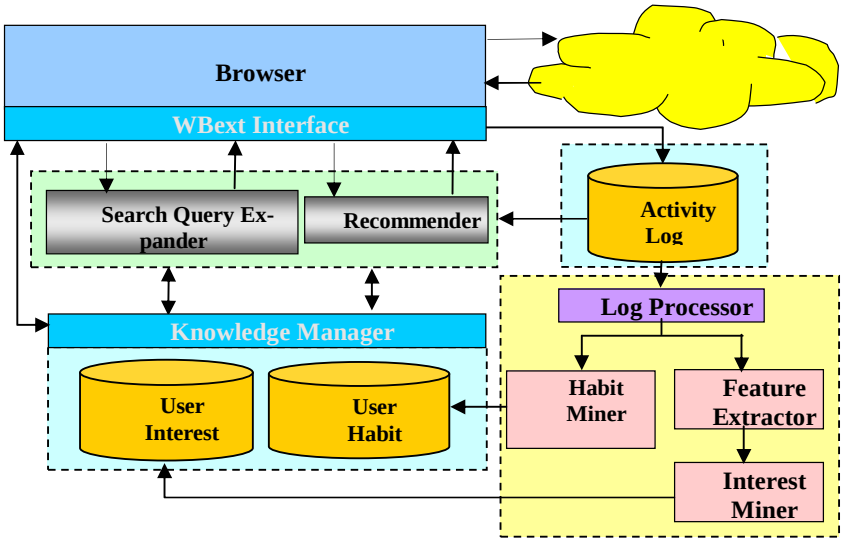

Figure 2. WBext: The System Architecture.

ests, and the Habit Miner mines the habit of the user. The resulting knowledge is sent to the Knowledge Base.

Two types of **WBext Agents**, a Search Query Expander and a Recommender, have been implemented in the system to make use of the mined knowledge. The search query expander expands a keyword-based search query into a number of queries that better reflect the user's requirements. The Recommender provides the user with a selected list of hyperlinks appeared in the Web page that the user is browsing in the order of relevancy based on both user interests and habits.

WBext works as follows. When a user uses the browser to surf the Web, her activities are captured. The two types of knowledge are mined and accumulated in the knowledge base. After the system accumulates sufficient knowledge, the search query expander will be active when the system detects that the user is issuing a keyword search to search engines, and the recommender will be active when the user browses a Web page. All the functions can be turned off so that WBext will work just as a plain browser.

2.2 Data preparation

The WBext interface produces an activity log for the activities carried out in each browser window. To prepare the activity logs into data that are digestible by data mining algorithms, three data preparation tasks are performed by the Log Processor: Timestamp Normalization, Focus History Generation, and Transaction Resolution.

Timestamp Normalization is to normalize the timestamps of all user activities to an absolute time value. This is because activities in different browser window sessions can be interrelated while the timestamps of activities stored in activity logs are

temporal values relative to the start time of a browser window session. To enable temporal comparison between activities of different activity logs, we need to use the normalized timestamps.

Another data preparation task, **Focus History Generation**, enables the system to have accurate information on how much time the user actually spends on each web page. Due to the existence of multiple browser windows, popup advertisements and frame sets, multiple pages can be opened simultaneously for browsing. Focus History is a sequential order of page focusing activities. It gives accurate information on how these pages are actually viewed by the user.

The last task is **Transaction Resolution**. To explore the World Wide Web for desired information, the user performs a series of activities, which are related to one other. Each series is considered as a transaction. The system tries to resolve the activities in logs to disjoint transactions.

2.3 Unsupervised Learning of Knowledge

One of the key features of the system is its capability of learning valuable knowledge from the client-side activity log without the user's supervision – the user does not need to assist the system in analyzing the activity data. Two types of knowledge, User Interest and User Habit, are learned by the system, using data mining algorithms.

2.3.1 Mining of User Interest

A user has a number of interest areas, which are reflected in the activities carried out in the transactions. To capture the characteristics of user interests, similar transactions are grouped together to form clusters. Each cluster represents one interest area of the user. The vector-based model is used for this clustering analysis process. Whenever a new transaction is performed by the user and its activities is captured by the system, important features are extracted from the transaction to produce a feature vector. A cluster in the user interest base is composed of transactions whose feature vectors are similar to each other.

2.3.2 Mining of User Habit

When browsing the web, users usually have some habits that are reflected in the associations among activities. For instance, while visiting news websites, the user tends to look at the technology and sports section. Such user browsing habits are represented by association rules of form $A \rightarrow B$, where A and B are sets of user activities. Such a rule indicates that, if the user's current activities are A, the next activity is most likely B. We developed a modified version of the Apriori Algorithm, which we call the Partial Apriori Algorithm. It can mine adaptable and generic rules from the activity data.

2.4 Application of knowledge

When a user uses WBext for a period of time, the knowledge base grows in depth and breadth. With a proper scale of knowledge, the system can aid the user by expanding search queries and offering link recommendations.

2.4.1 Search Query Expansion

Common users use only a few keywords in their search queries, which often fail to precisely describe to the search engines what the users really want. As a result, most search attempts produce a large number of "matched" URLs. The user needs to decide the relevance manually by browsing through the result list, or to navigate deeper following the links in the result list.

When the user submits a simple keyword query to a search engine through WBext, the system will try to identify the user's current interest against the interest areas in the User Interest Base. If the activities in the current session are similar enough to one of the interest clusters, the system expands the original search query to multiple extended queries, each of which is produced by adding extra query terms to the original query. Such terms are extracted from the feature vectors in the identified interest cluster.

The results from the extended search queries are merged together and re-ranked to produce a list of extended search results. This list has improved precision as more specific keywords are added; therefore, the search attempt becomes more focused on the user's interest. Furthermore, the recall of a search attempt is also improved as a wider variety of vocabulary is introduced. Finally, since the ranking of the list is also based on the user's interest, the user will find the ranking of expanded query results closer to their preference.

2.4.2 Link Recommendation

When the user is viewing a web page with a large number of hyperlinks, the system will recommend a few links that are expected to be more relevant to the user. Each link is given a ranking score, and the links whose scores are above a threshold form the list of recommendation.

The ranking is divided into habit-based ranking and interest-based ranking. To enable habit-based ranking, the system keeps comparing the activities with the rules in the user habit base during the course of web browsing. Links that match the rules in the user habit base are given high scores. For interest-based ranking, terms in the links are compared to the terms and weights in the feature vectors within the identified cluster. Scores are then given to links according to their corresponding weights. The system combines habit-based ranking and interest-based ranking for recommending primary links.

3. Experimental Results

We have conducted preliminary experiments to evaluate the system. The browser extensions are implemented in Visual C++ as a Browser Helper Object (BHO) of the

Microsoft Internet Explorer. The experiments were done on an Intel Pentium III 800Mhz machine with 512 MB memory. The operating system was Microsoft Windows 2000 Professional. In all of the experiments, the browsing speed of WBext was comparable to that of a plain browser (the difference was hardly noticeable). In this section, we show results concerning the effectiveness of knowledge discovery, search query expansion, and link recommendations.

3.1 On Knowledge Discovery Algorithms

To show the effectiveness of knowledge discovery capability of our system, we performed an experiment during which the user browsed a number of pages related to the interests of "Java", "News" and "Data Mining" in an interleaved fashion (Table 1).

Table 1. The user browsing sessions for mining

Session	Interest	Description
01	Java	Java certification information
02	News	News, mainly technology, little sports
03	Java	Java servlets
04	News	News, sports, science and technology
05	Java	Java certification mock papers
06	Java	Java tutorials in J2EE and JDBC
07	News	News of major IT firms
08	News	Sports news
09	News	News on IT, NBA
10	News	Sports news, NBA
11	Java	Preparation of certification exams, mainly Java
12	Java	Java online tutorials
13	Data Mining	General data mining research topics
14	Data Mining	General knowledge discovery research topics
15	News	General sports news
16	Java	Java server side programming
17	News	General news browsing
18	News	News from portal sites, from different content providers
19	News	News from portal sites, from different content providers
20	Data Mining	Clustering analysis algorithms
21	Data Mining	SIGKDD conferences and papers
22	Data Mining	Theories in data mining in large databases
23	News	Cable news
24	Data Mining	Web mining
25	Java	Java sample code
26	Data Mining	Association rule mining
27	Java	Java Server Pages (JSP)
28	Data Mining	Client side web mining
29	Java	Comparison of J2EE and .Net
30	Java	JDBC and JDO

The system captured the browsing activities and identified a number of transactions. The interest miner grouped the transactions into clusters, as shown in Figure 3. It can be clearly seen that most transactions are correctly grouped into the clusters representing the interests. Nevertheless, there are a few transactions scattered around the main clusters due to the large variance in the features of the transactions. This is most prominent in the transactions of the News interest.

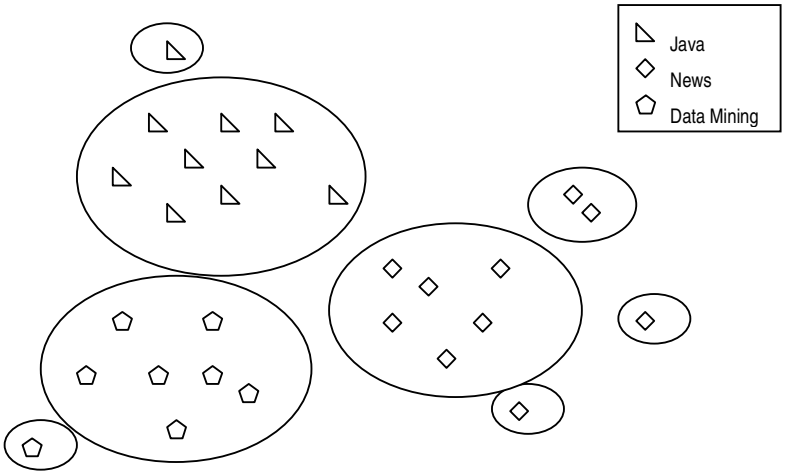

Figure 3. Clustering user sessions

3.2 On Search Query Expansion

We then performed search query expansion experiments based on the knowledge discovered from the previous experiment. The user submitted a simple search keyword query for each topic, and the system used the user interest knowledge to perform search query expansion. The precision of a search query is defined as the percentage of the results that match the user's interest. The effectiveness of search query expansion is measured by the ratio of the precision of the first 20 expanded search results to that of the first 20 original search results. The experiments show (Tables 2 and 3) that search query expansion is capable of improving precision of a search attempt by a ratio of 2 to 3.4.

For example, an original search query was "Java Book" and the precision of the first 20 matches returned from a search engine (Google) was 25%. WBext expanded the query into five queries and sent them to the same search engine, each with one of the following terms identified from the user interest knowledge base: "Certification", "Enterprise", "J2EE", "Servlet", and "JDBC". The precision of the first 20 matches of the combined and re-ranked expanded query results was 85%, which led to a precision ratio 3.40.

Table 2. The original search queries in search expansion experiments

Search query	Interest	Precision of the first 20 results
Java Book	Java	25%
Data Mining example	Data Mining	35%
News Archive	News	20%

Table 3. The expanded search queries in search expansion experiments

Original query	Expanded query terms	Precision of first 20 results
Java Book	Certification Enterprise J2EE Servlet JDBC	85%
Data Mining example	Algorithm Conference Large-Scale Web	75%
News Archive	Technology Sports Hacker NBA Entertainment	40%

3.3 On Link Recommendation

The last set of experiments assessed the effectiveness of link recommendation by measuring the precision and recall of the recommendation lists. The precision was defined as the percentage of recommended links that match the interests of the user. The recall was defined as the percentage of interesting recommended links to the interesting links in the web page.

We performed four web browsing sessions A, B, C, and D (Table 4). In each session, the user first conducted a number of initial browsing activities and then started to get recommendations of links from the system when encountering web pages with a large number of hyperlinks in them. We evaluated the effectiveness of three recommendations for each session.

Session A was random browsing with no interest areas; therefore, only habit-based recommendation was performed. Sessions B, C, and D focused on Java programming readings, news headlines of the day, and Data Mining reading materials, respectively. Both user interests and habits were used for recommendation in sessions B, C, and D.

As shown in Figures 4 and 5, the precision of the link recommendations was high (80-100%) and the recall varied from 20% to 94%. The random session A had a high precision value (88-100%) due to the reliability of user habits, while it suffered from a lowest recall value (20%) in the first recommendation. The effectiveness of session C was lower than that of sessions B and D, due to the diversity of terms used in news

headlines. The effectiveness of sessions B was especially encouraging because the topics had distinctive terms such as “JDBC”, “Servlet”, and others.

Table 4. The web browsing sessions in link recommendation experiments

Session	Interest	Description of initial activities
A	None	Browsing through various interest areas
B	Java	Reading pages about Java programming
C	News	Looking for the news headlines of the day
D	Data Mining	Reading pages about data mining techniques

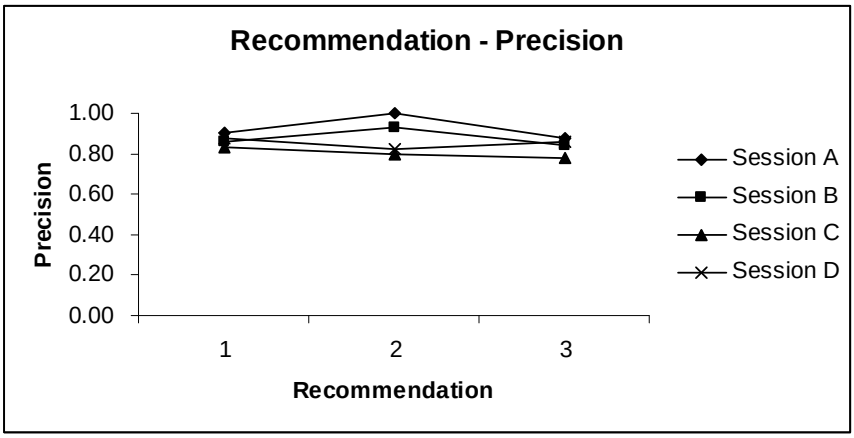

Figure 4. Precision of link recommendations

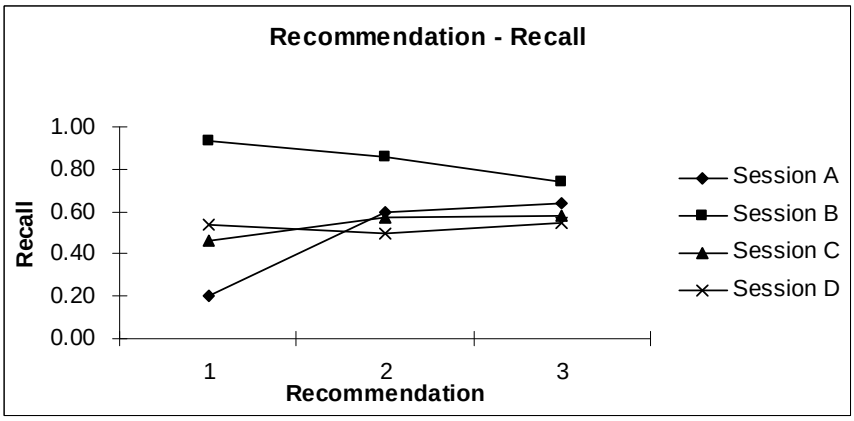

Figure 5. Recall of link recommendations

4. Related Work

Work related to ours comes from three areas: collaborative filtering, web log mining, and intelligent agents.

Collaborative filtering, or collaborative personalization, is a technique to tailor content for specific users based on some collaborative measures that requires participation of the users. The web content provider use this means of personalization extensively. Examples include portal sites [15], online radio [5], news [8], and online learning [4]. Such systems keep user profiles and perform recommendations in the form of a personalized starting point, which straightly directs the user to the location of proper contents. In addition to recommendations, the profiles are also used for purposes such as pushing appropriate advertisements. In these systems, each profile has to be constructed with some form of collaboration with the user. In contrast, WBext does not require any collaborative efforts from the user.

Web log mining is an application of data mining techniques to discover knowledge from HTTP access logs of web servers. Similar to our client side mining approach, it requires pre-processing [3] steps on the logs and discovers patterns through association rules mining and clustering ([16], [7]). Nevertheless, our activity logs contain much more detailed and accurate information than web server logs, and our purpose is to improve individual user experience with all web sites that they access rather than improving individual web sites for the users that access those sites.

WBext is in the category of intelligent agents that provide users assistance in accessing the Web. Most intelligent agents ([6], [9]) run on the server side or depend on a server side component [2], which may cause privacy concerns. Moreover, they only track link following and web page visits as the raw data for learning tasks. In comparison, WBext monitors a wider variety of user activities and enhances the quality of knowledge discovered.

5. Conclusions

We have presented a client side mining approach to personalizing web browsing and search. A detailed user activity log on the client side is captured by monitoring the interactions between the user and the web browser. Within the activity log is precise information on how the user browses the web and the topics she/he is interested in. The activities log is resolved into disjoint transactions, each of which contains activities carried out for one specific interest area only. Disjoint clusters of transactions form the user interest base. In addition, association rules are mined among the activities carried out in each transaction. These rules are a formal representation of the user's web browsing habits.

Using the mined knowledge, two types of assistance are offered to the user: search query expansion and link recommendation. Search query expansion is activated when the user initiates a search attempt when he/she browses the web. A simple keyword query specified by the user is expanded to multiple extended queries, which are compositions of the original query terms and additional terms extracted from the cluster

that represents the current interest of the web browsing session. Link recommendation is provided whenever the user visits a web page. Both the user interest and user habit knowledge can be utilized for this purpose.

We have evaluated the effectiveness of our approach on WBext, our prototype system. The initial results of the experiments are encouraging. Future work on the client side activity log mining approach includes finer distinction of topics within the same interest area hierarchically and generation of association rules that applies to particular interest areas.

References

1. A. G. Buchner, M. Baumgarten, S. S. Anand, M. D. Mulvenna, and J. G. Hughes: User-Driven Navigation Pattern Discovery from Internet Data. WebKDD, 1999.
2. L. Chen and K. Sycara: WebMate: Personal Agent for Browsing and Searching. In Proc. 2nd Int'l Conf on Autonomous Agents, pp. 132-139. 1998.
3. R. Cooley, B. Mobasher, and J. Srivastava: Data Preparation for Mining World Wide Web Browsing Patterns. Journal of Knowledge and Information Systems, 1(1), 1999.
4. C. Groeneboer, D. Stockley, and T. Calvert: Virtual-U: A collaborative model for online learning environments. In Proc. Second International Conference on Computer Support for Collaborative Learning, Toronto, Ontario, December 1997.
5. D. B. Hauver and J. C. French: Flycasting: Using Collaborative Filtering to Generate a Playlist for Online Radio. International Conference on Web Delivering of Music, 2001.
6. T. Joachims, D. Freitag, and T. Mitchell: WebWatcher: A tour guide for the World Wide Web. In Proc. IJCAI-97, Nagoya, Japan.
7. A. Joshi and R. Krishnapuram: On Mining web Access Logs. In Proc. 2000 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery 2000, pp. 63--69, 2000.
8. K. Lang: NewsWeeder: Learning to Filter Netnews. In Proc. 12th International Conference on Machine Learning, pp. 331-339, 1995.
9. J.P. McGowan, N. Kushmerick, and B. Smyth: Who do you want to be today? Web Personae for personalized information access. International Conference on Adaptive Hypermedia and Adaptive web-Based Systems (AH2002), Malaga, Spain.
10. D. Mladenic: Personal WebWatcher: Implementation and Design. Technical Report, IJS-DP-7472, Dept. of Intelligent Systems, J. Stefan Institute, Slovenia, 1996.
11. B. Mobasher, H.Dai,T.Luo,M.Nakagawa,Y.Sun, and J.Wiltshire: Discovery of Aggregate Usage Profiles for Web Personalization, WebKDD 2000.
12. M.D. Mulvenna, S.S. Anand, and A.G. Buchner: Personalization on the Net Using Web Mining. CACM, Vol. 43(8):pp. 123--125, August 2000.
13. R. Rafter, B. Smyth: A domain analysis methodology for collaborative filtering. 23rd BCS European Annual Colloquium on Information Retrieval Research, Darmstadt, Germany, April 2001.
14. C. Shahabi: Knowledge Discovery from users web-page navigation. ICDE-RIDE 1997.
15. Yahoo! Inc.: Welcome to My Yahoo! http://my.yahoo.com
16. O. R. Zaïane, Man Xin, Jiawei Han: Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs. ADL 1998: 19-29.