Web Recommendations for Mobile Web Browsing

Anupam Thakur
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202
anupam.thakur@und.edu

Wen-Chen Hu
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202
wenchen@cs.und.edu

Naima Kaabouch
Department of Electrical
Engineering
University of North Dakota
Grand Forks, ND 58202
naimakaabouch@mail.und.edu

Liang Cheng
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202
vikingcheng@gmail.com

Abstract

Mobile phone usage is growing very fast and the number of mobile phone users is increasing day by day. Smartphones, a kind of cellular phones, allow mobile users to browse the mobile World Wide Web. It is believed that in the near future the mobile handheld devices will become the standard clients for Web access. However, the implementation of mobile Internet access is a challenging task because of various factors like small screens of mobile devices, low data transmission rate, awkward input and output methods, and low battery life. These problems must be solved before the smartphones can be used to browse the mobile Web seamlessly, easily, and effectively. This research investigates a new method for enhancing mobile Web access by using World Wide Web usage mining. The proposed system includes five major components: (i) usage data gathering, (ii) usage data preparation, (iii) usage navigation pattern discovery, (iv) usage pattern analysis and visualization, and (v) usage pattern application of mobile Web browsing. Whenever a mobile user visits a Web site, the discovered sequences are then used to find recommended or related links, which are inserted into the top of each requested page. These Web recommendations can significantly enhance the mobile-browsing experience because the aggregated access behavior of all users usually provides useful, effectively browsing information for mobile users.

1 Introduction

The number of smartphones shipped is increased fast in recent years and it is expected the number of smartphones shipped will surpass the number of PC shipped in the near future. When people started using handheld devices to browse the mobile Internet about ten years ago, Webmasters usually created two versions of their Web pages. One version using HTML is for desktop browsers and the other one using WML, cHTML, or other languages is for microbrowsers. However, this approach has been proved futile and time-consuming and most Web sites have only one version in HTML for both desktop browsers and microbrowsers today. Most Web pages are mainly designed for desktop or notebook computers. They usually do not suit the devices. This research proposes a World Wide Web recommendation system for enhancing mobile Web access. Whenever a mobile user visits the Web site, the discovered sequences then used to find the recommended or related links which might help the user in easy browsing. This enhances the mobile-browsing experience because the aggregated access behavior of all users usually provides useful, effective browsing information for mobile users.

This project first collects the mobile Web usage information and then implements a Web usage mining on the collected information for creating a global recommender system. A Web proxy is used to collect Web usage data, which is then stored in a database. The data is queried and further refined by the recommender system to provide the global usage metrics and statistics including the most popular paths, numbers of requests, and most popular resources. This approach will improve the browsing experience for mobile users. There are several issues to be considered in designing a Web usage model for the Web browsing. This paper will consider the topics relating to (i) usage data gathering, (ii) usage data preparation, (iii) navigation pattern discovery, (iii) pattern analysis and visualization, and (v) pattern applications.

This article investigates a method for enhancing mobile Web accesses by using World Wide Web usage mining. It provides Web recommendations for mobile users and can significantly enhance the mobile-browsing experience. The rest of the article is organized as follows. Section 2 discusses related Web usage mining. The proposed mobile Web usage mining system is introduced in Section 3. Some experimental results are given in the Section 4. The final section includes the conclusion part and the related future work.

2 Related Research

World Wide Web Data Mining includes content mining, hyperlink structure mining, and usage mining. All three approaches attempt to extract knowledge from the Web, produce some useful results from the knowledge extracted, and apply the results to certain real-world problems. The first two apply the data mining techniques to Web page contents and hyperlink structures, respectively. The third approach, Web usage mining, the theme of this article, is the application of data mining techniques to the usage logs of large Web data repositories in order to produce results that can be applied to many practical subjects, such as improving Web sites/pages, making additional topic or product

recommendations, user/customer behavior studies, etc. It is necessary to examine what kind of features a Web usage mining system is expected to have in order to conduct effective and efficient Web usage mining, and what kind of challenges may be faced in the process of developing new Web usage mining techniques. A Web usage mining system should be able to:

- Gather useful usage data thoroughly,
- Filter out irrelevant usage data,
- Establish the actual usage data,
- Discover interesting navigation patterns,
- Display the navigation patterns clearly,
- Analyze and interpret the navigation patterns correctly, and
- Apply the mining results effectively.
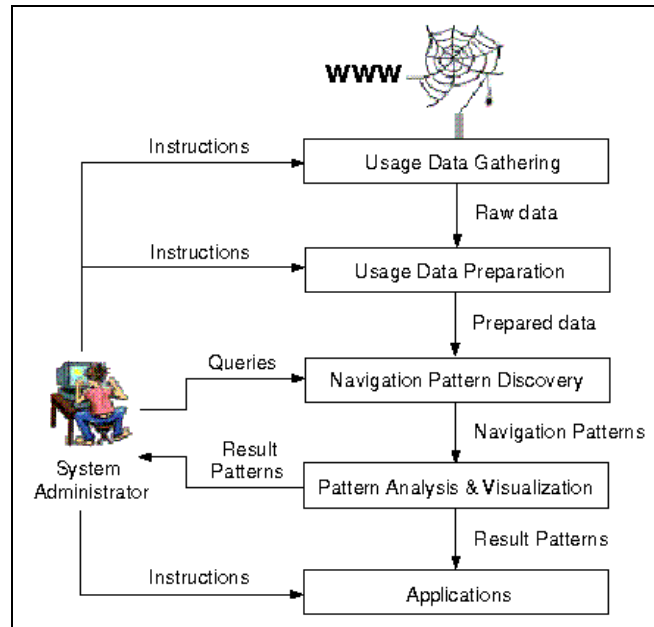
Figure 1: A Generic Structure of Web Usage Mining Systems.

A variety of implementations and realizations are employed by Web usage mining systems. This section gives a generic structure of the systems as shown in Figure 1, each of which carries out five major tasks (Hu, Zong, Chu, & Chen, 2002; Clifton, Cooley, & Rennie, 2004; Cooley, Mobasher, & Srivastava, 1999):

- *Usage data gathering*: Web logs, which record user activities on Web sites, provide the most comprehensive, detailed Web usage data.
- *Usage data preparation*: Log data are normally too raw to be used by mining algorithms. This task restores the users' activities that are recorded in the Web server logs in a reliable and consistent way.
- *Navigation pattern discovery*: This part of a usage mining system looks for interesting usage patterns contained in the log data. Most algorithms use the method

of sequential pattern generation, while the remaining methods tend to be rather ad hoc (Xing & Shen, 2004; Agrawal & Srikant, 1995; Agrawal & Srikant, 1994).

- *Pattern analysis and visualization*: Navigation patterns show the facts of Web usage, but these require further interpretation and analysis before they can be applied to obtain useful results.
- *Pattern applications*: The navigation patterns discovered can be applied to the following major areas, among others: (i) improving the page/site design, (ii) making additional product or topic recommendations, (iii) Web personalization, and (iv) learning the user or customer behavior (Resnick & Varian, 1997).

A Web usage mining system can be further divided into the following two parts (Eirinaki & Vazirgiannis, 2003; Spiliopoulou, 2000):

- *Personal*: A user is observed as a physical person, for whom identifying information and personal data/properties are known. Here, a usage mining system optimizes the interaction for this specific individual user. Personal systems are actually a special case of impersonal systems. You can easily infer the corresponding personal systems, given the information for impersonal systems.
- *Impersonal*: The user is observed as a unit of unknown identity, although some properties may be accessible from demographic data. In this case a usage mining system works for a general population.

The useful information gathered from the data preparation stage can be used as an input for various usage mining algorithms such as sequential pattern discovery, Association rule discovery, Data clustering, Data classification, sequential analysis and more. The results generated by this data mining process can be applied in many practical subjects such as studying user behavior, improve Web page design, recommending related useful information according to his/her profile, etc.

3 The Proposed Mobile Web Usage Mining System

Web mining refers to the overall process of discovering potential useful and previously unknown navigational information or knowledge from the Web data. Web usage mining is the procedure where the information stored in the Web server logs is processed by applying data mining techniques to extract statistical information and discover interesting usage patterns, cluster the user into groups according to their navigational behavior, and discover potential correlation between Web pages and user groups. Therefore, the requirement for predicting user needs to improve the usability and user retention of a Web site can be addressed by personalizing it.

3.1 The System Structure

The structure of a Web usage mining process is divided into different sections. Figure 2 graphically represents the structure of the Web usage mining process. The Web usage mining process is done step by step, from usage data collection to pattern discovery and its implementation. Details of the system structure will be given in the rest of this paper.
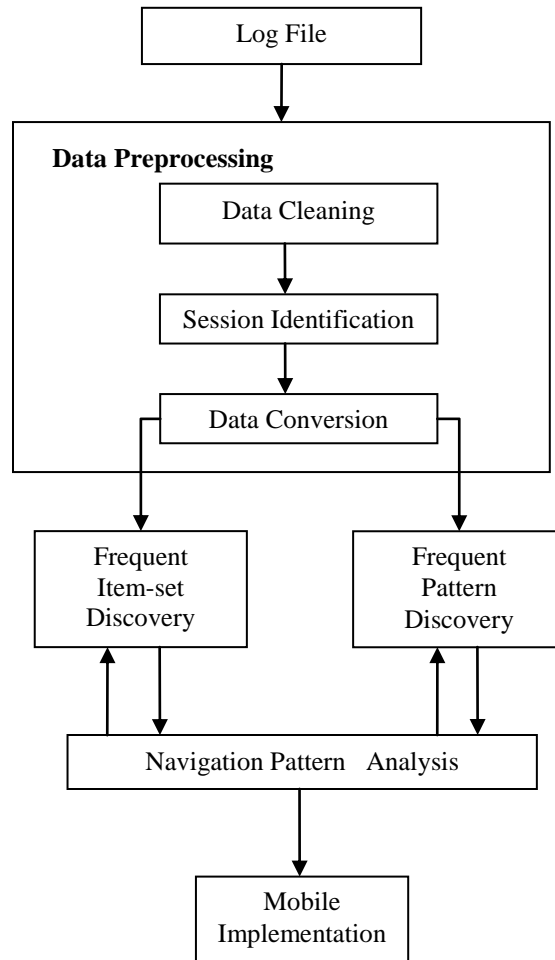
```
                    ┌─────────────────┐
                    │    Log File     │
                    └────────┬────────┘
                             │
                             ▼
        ┌────────────────────────────────────────────┐
        │ Data Preprocessing                         │
        │         ┌─────────────────────┐            │
        │         │   Data Cleaning     │            │
        │         └──────────┬──────────┘            │
        │                    │                       │
        │                    ▼                       │
        │         ┌─────────────────────┐            │
        │         │ Session Identification           │
        │         └──────────┬──────────┘            │
        │                    │                       │
        │                    ▼                       │
        │         ┌─────────────────────┐            │
        │         │   Data Conversion   │            │
        └─────────┴─────────────────────┴────────────┘
```

Figure 2: The Web Usage Mining Process Used in This Project.

3.2   Web Usage Data Collection

The Web usage mining is a process involving not just the use of pattern discovery algorithms, but also selecting the best way to capture important transactional data. In order to gather the intelligence from the Web, it is important to know what kinds of data are available and what programs are most effective for collecting this information. A Web log is a record of the HTTP transactions performed by Web software components. The Web logs actually capture activities from online users and exhibit a wide range of different behavioral patterns. Analysis of Web logs is useful for understanding the characteristics of users' behavior and discovering some patterns. Web usage data collected from different sources reflects the different types of usage tracking appropriate for different purposes. Server statistics provided by the Web log analysis tools provides metrics for evaluating the success of the server in serving pages to users. A common log file (W3C, n.d.) is created by the Web server to keep track of the requests that occur on a

Web site. When a page is requested, the Web proxy downloads the page source and, finds the embedded links, and reroutes them through itself. The requested URL is passed as an environment variable and is used for logging, so that each resource request is logged with its source resource and its target resource. Figure 3 is an example of the functionality provided by the Web proxy developed for this research (Hong & Landay, 2001).
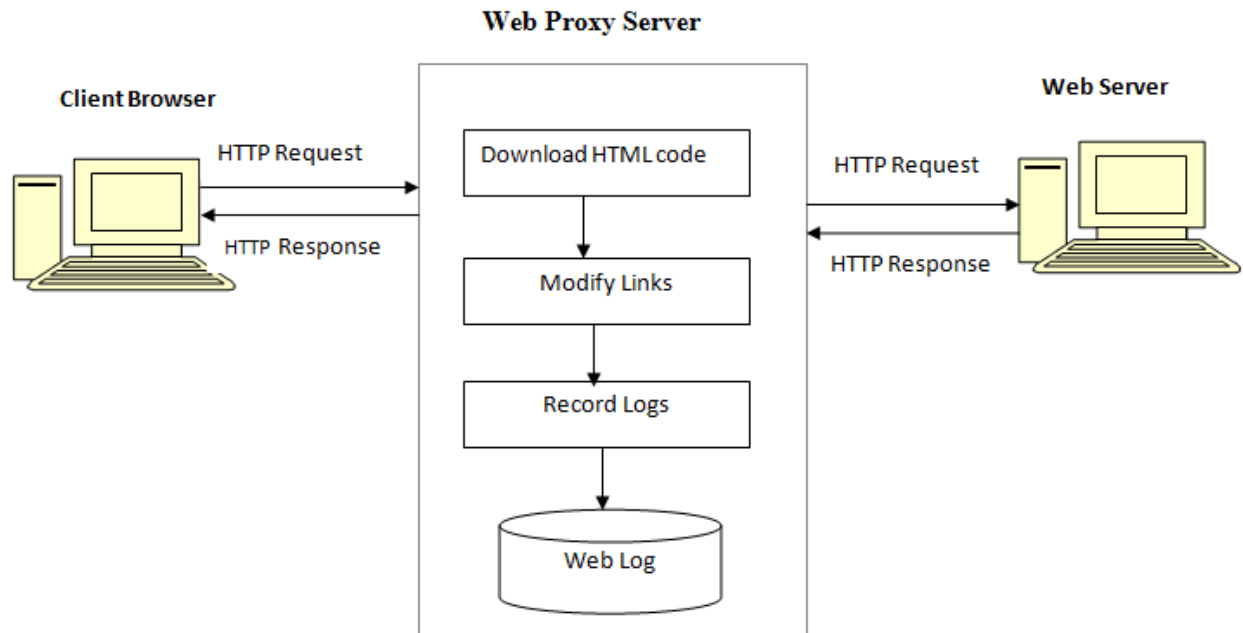
Web Proxy Server

| | Client Browser | | Web Server |

Figure 3: A System Structure of the Web Proxy Server.

The proxy implementation for this thesis uses CGI-Perl technology. Decoupling the scripts from the Web server requires a well-defined interface for passing data between the two pieces of software. URL's that correspond to scripts typically includes a '?' character. Programming languages, such as Perl, have functions that returns the value of a given environment variable. The Web server provides a variety of information (CGI environment variables) to the script. This information is related to the server, client, and request. The proxy used here is different from the traditional Web proxies, where the traditional Web proxies serve as a relay point for all of a user's Web traffic, and the user's browser must be configured to send all the requests through the proxy. The proxy developed for this project is a URL-based proxy, similar to WebQuilt or WebSIFT. A UEL-base proxy accepts as a input a URL, redirects all the links so that the subsequent URL's point to the proxy with the intended destination encoded in the URL's query string.

Environmental variables (Perlfect Solutions, n.d.) are a series of hidden values that the Web server sends to every CGI you run. Environment variables are stored in a hash called "%ENV." The variables shown in Table 1 are useful for log data gathering.

5

Name	Description
REMOTE_ADDR	The IP address of the remote host making the request
REQUEST_URI	A URI that provides an address of a server object
HTTP_REFERER	The URL of the page that called your script
REMOTE_HOST	The hostname making the request. If the server does not have this information, it should set REMOTE_ADDR and leave this unset
QUERY_STRING	The information which follows the '?' in the URL which referenced this script. This is the query information. It should not be decoded in any fashion. This variable should always be set when there is query information, regardless of command line decoding
SERVER_NAME	The server's hostname, DNS alias, or IP address as it would appear in self-referencing URLs
REQUEST_METHOD	The method with which the request was made. For HTTP, this is "GET," "HEAD," "POST," etc.
SERVER_SOFTWARE	The name and version of the information server software answering the request (and running the gateway). Format: name/version
CONTENT_LENGTH	The length of the said content as given by the client
SERVER_PORT	The port number to which the request was sent
SCRIPT_NAME	The interpreted pathname of the current CGI (relative to the document root)
CONTENT_TYPE	For queries which have attached information, such as HTTP POST and PUT, this is the content type of the data
CONTENT_LENGTH	The length of the said content as given by the client
HTTP_USER_AGENT	The browser the client is using to send the request. General format: software/version library/version
REMOTE_ADDR	The IP address of the remote host making the request
SERVER_PROTOCOL	The name and revision of the information protocol this request came in with. Format: protocol/revision
GATEWAY_INTERFACE	The revision of the CGI specification to which this server complies. Format: CGI/revision

Table 1: The Environment Variables.

To help the mobile user find relevant pages fast, the recommender system suggests the user with the recommendations on top of the page as shown in Figure 4
Figure . The mobile user with the help of recommendations can directly jump to the destination page. The recommender system generates recommendations depending on the users' past browsing history. The idea behind the recommender system is to construct the recommendations for the new mobile users with no browsing history.

You may like these links
→HTML Tutorial
→HTML Getting Started
→CSS Tutorial
→HTML Styles

What Other People are Browsing
→HTML Tutorial
→Introduction to HTML
→HTML Getting Started
→CSS Tutorial
→HTML Basic
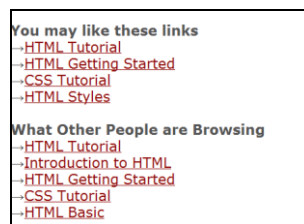
Figure 4: Recommendations on Top of a Page.

When a page is requested, the Web proxy captures the page source, finds the embedded links, and re-routes them through itself. The requested URL is passed as an environment variable and is used for logging that each resource request is logged with its source resource – the resource which the request is made – and its target resource – the resource being requested. Figure 5 shows the original HTML source and Figure 6 demonstrates the results – the redirection of links of the proxy.

```
<tr>
 <th><a href="http://www.cs.und.edu/~wenchen/#basic">Basic Information</a></th>
 <th><a href="http://www.cs.und.edu/~wenchen/#bio">A Short Bio</a></th>
 <th><a href="http://www.cs.und.edu/~wenchen/#education">Education</a></th>
 <th><a href="http://www.cs.und.edu/~wenchen/#experience">Experience</a></th>
</tr>
```

Figure 5: An Example of HTML Source Script.

```
<tr>
 <th>
  <a href="http://www.cs.und.edu/~thakur/cgi-
bin/thesis/WebMining.cgi?http://www.cs.und.edu/~wenchen/#basic">
   Basic Information
  </a>
 </th>
 <th>
  <a href="http://www.cs.und.edu/~thakur/cgi-
bin/thesis/WebMining.cgi?http://www.cs.und.edu/~wenchen/#bio">
   A Short Bio</font>
  </a>
 </th>
 <th>
  <a href="http://www.cs.und.edu/~thakur/cgi-
bin/thesis/WebMining.cgi?http://www.cs.und.edu/~wenchen/#education">
   Education
  </a>
 </th>
 <th>
  <a href="http://www.cs.und.edu/~thakur/cgi-
bin/thesis/WebMining.cgi?http://www.cs.und.edu/~wenchen/#experience">
   Experience
  </a>
 </th>
</tr>
```

Figure 6: The Proxy Modified Script of the HTML Script in Figure 5.

For logs that span long periods of time, it is very likely that individual users will visit the Web site more than once or their browsing may be interrupted. The goal of session identification is to divide the page accesses of each user into individual sessions. A time threshold is usually used to identify sessions. Ideally, each user session gives an exact accounting of who accessed the Web site, what resources were requested and in what order, timestamp of each resource viewed. The proxy built for this thesis records all the

navigational data from which sessions are created. Each time the user starts using the proxy, it is considered as a session which is recorded in the log by the time it is viewed. A sample of Web logs saved in Oracle11g is shown in the Table 2. In the table, each entry is a record of all environment variables which the server generates as a part of handling the user request. By analyzing the Web usage data, Web mining system discovers the useful knowledge about the system's usage characteristics and user's interest which is applied to generate recommendations for mobile users.

Raw Web Log							
REQUESt_ID	DAY	TIME	FROM_URL	TO_URL	REMOTE_IP	SERVER_PROTOCOL	GATEWAY_INTERFACE
13367	Dec 2, 2009	1259812826	START	http://www.cs.und.edu/	172.20.4.156	HTTP/1.1	CGI/1.1
13368	Dec 2, 2009	1259812831	http://www.cs.und.edu/	http://www.cs.und.edu /Home/department_information.aspx	172.20.4.156	HTTP/1.1	CGI/1.1
13369	Dec 2, 2009	1259812835	http://www.cs.und.edu /Home/department_information.aspx	http://www.cs.und.edu /Home/why_major_in_cs.aspx	172.20.4.156	HTTP/1.1	CGI/1.1
13370	Dec 2, 2009	1259812840	http://www.cs.und.edu /Home/why_major_in_cs.aspx	http://www.cs.und.edu/People /Default.aspx	172.20.4.156	HTTP/1.1	CGI/1.1
13371	Dec 2, 2009	1259812855	START	http://www.und.edu/	172.20.4.156	HTTP/1.1	CGI/1.1
13372	Dec 2, 2009	1259812872	http://www.und.edu/	http://www.und.edu/currentstudents.jsp	172.20.4.156	HTTP/1.1	CGI/1.1
13373	Dec 2, 2009	1259812881	http://www.und.edu /currentstudents.jsp	http://www.und.edu/academics.jsp	172.20.4.156	HTTP/1.1	CGI/1.1
13374	Dec 2, 2009	1259812891	http://www.und.edu/academics.jsp	http://www.und.edu /prospectivestudents.jsp	172.20.4.156	HTTP/1.1	CGI/1.1
13375	Dec 2, 2009	1259813014	START	http://m.yahoo.com	80.239.242.51	HTTP/1.1	CGI/1.1
13376	Dec 2, 2009	1259813029	http://m.yahoo.com	http://m.yahoo.com/p/sports	80.239.242.51	HTTP/1.1	CGI/1.1
13377	Dec 2, 2009	1259813047	http://m.yahoo.com/p/sports	http://m.yahoo.com/w/sports /all/?.ts=1259813032&.intl=US&.lang=en	80.239.242.51	HTTP/1.1	CGI/1.1
13378	Dec 2, 2009	1259813071	http://m.yahoo.com/w/sports/all/	http://m.yahoo.com/w/yffl /?.intl=US&.lang=en	80.239.242.51	HTTP/1.1	CGI/1.1

Table 2: An Example of Web Log Data Stored in a Database.

4 Experimental Results

In order to evaluate the effectiveness of the proposed Web recommender system and demonstrate the capability of the system, we conduct experiments on the data sets and summarize the results in this section.

5.1 Experimental Setup

The proposed system uses Oracle11g and all programs are hosted by a Linux server. JDBC is used to connect Java to the Oracle11g. Perl, CGI, Java are the main programming languages. The system details are listed in the Table 3.

Software	Model/Version/Type
Server	Linux
Web server	Apache/2.0.46 (Red Hat) Server
Database server	Oracle 11g
Languages used	Perl, CGI, JDBC, XHTML and Java
Microbrowser	Opera mini simulator
Other tools	Putty, WS-FTP, Oracle SQL developer 1.5.5, Graphviz-graphic visualization software, etc.

Table 3: System Information.

In this section, several experiments are conducted to evaluate the performance of our proposed system on the collected datasets. The tests will be conducted on the gathered data and the experimental results will be displayed. There are some limitations with the system. For example, it will not support the sites which employ Javascript, VBscript, Ajax, or Java servlets, sites with forms backed by common gateway interface (CGI), Active Server Pages (ASP), or Java2 Enterprise Edition (J2EE) servlets and server pages. Due to the nature of the Web proxy server, Web pages which do not make traditional Web resource requests to retrieve new pages cannot be captured by the Web proxy and thus do not appear in the Web server logs. Given that they do not appear in the database, accurate recreation of the navigation paths is impossible. Therefore, a single Web site using traditional Web resources has been chosen for this experiment so that all requests for resources can be captured, logged and mined. The schema of the database table for the log data is given in Table 4.

Name	Type
REQUEST_ID	NOT NULL NUMBER(5)
DAY	VARCHAR2(30)
TIME	NUMBER(15)
FROM_URL	VARCHAR2(100)
TO_URL	VARCHAR2(100)
REMOTE_IP	VARCHAR2(50)

SERVER_PROTOCOL	VARCHAR2(20)
GATEWAY_INTERFACE	VARCHAR2(20)
SERVER_PORT	NUMBER(5)
SCRIPT_NAME	VARCHAR2(100 CHAR)
HTTP_USER_AGENT	VARCHAR2(200 CHAR)

Table 4: The Log Table Used by This Project.

5.2 Experiments

The proposed recommendation system uses the Web site http://www.w3schools.com/ as the testing site. The Web proxy is a Perl CGI script that handles user Web resource requests by querying the user for a base site address on its opening screen and then routing subsequent user Web site resource request through itself by modifying the embedded hyperlinks. Figure 7 shows the entry page for the proposed system.
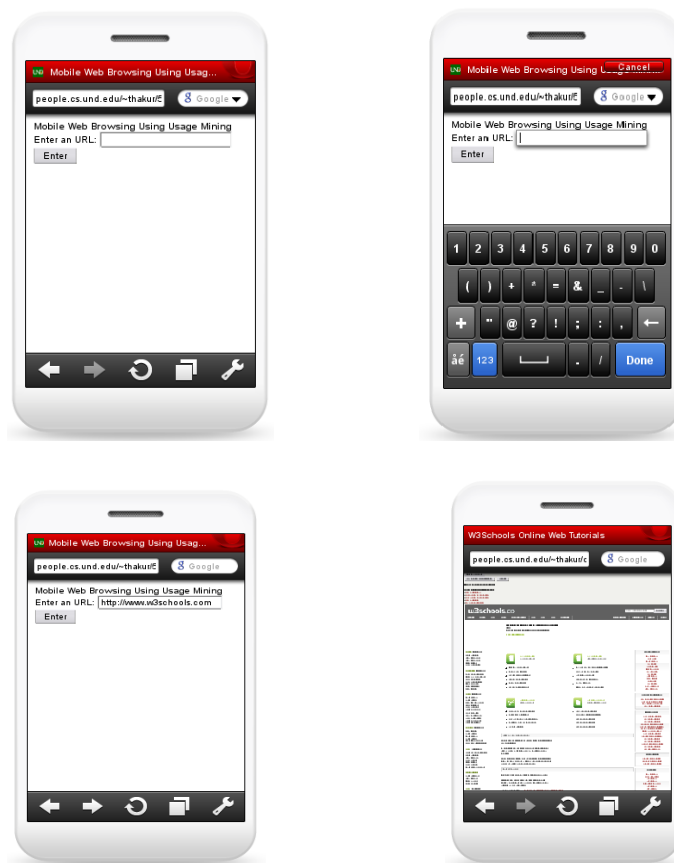
Figure 7: The Entry Page for the Proposed System.

When a new user starts using the system, the browsing information is empty. So it is difficult to provide him/her with the personal recommendations. Figure 8 displays the generic recommendations when a new user starts using the system.
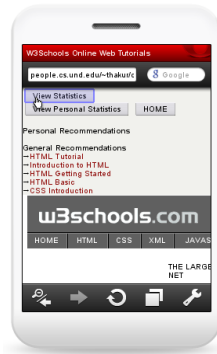
Figure 8: Recommendations with No Historical Records Accessed.

The generic recommendations displayed for the new user are based on the other users' browsing data. If the user used the system in the past, then both the personal and generic recommendations are shown to the user as shown in from Figure 9.
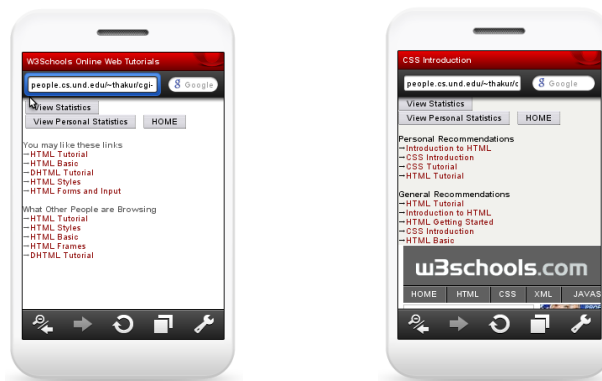
Figure 9: Recommendations with Historical Records Accessed.

The input for the Web usage mining process is referred to as the prepared log data, that gives an exact account of who accessed the Web site and in what order. It logs all the user activities on the Web site. The information contained in the raw Web log does not reliabley represent an authentic user session. The raw Web log needs to be filtered so that all the unnecessary and irrelevant information about requests and accesses from the robots (Search engines and other programs) can be seperated. The idea of recommender systems is to automatically suggest items to each user that they may find appealing. Server gathers information regarding the user by analyzing the choices, decision and

11

maneuvers the user makes. The Web logs are grouped by the session_id and visitor's IP address and then all entries with same IP address are ordered by their access time. A huge number of accesses(hits) are registered and collected in an ever-growing Web log. Each user session induces a user trail through the site. During the browsing session, user followes a sequence of Web pages also refferred as "trail." Based on the Web user trails, Web access sequences are generated as shown in Table 5. Recommender system includes lots of data displayed in various forms. When analysing statistics, it is important to know what you are looking for and to separate important information from the data. Table 6 gives the support for each Web page visited.

Sequence	Pages in a Sequence
1	W3Schools Online Web Tutorials
	HTML Tutorial
	HTML Styles
	HTML Frames
2	W3Schools Online Web Tutorials
	HTML Tutorial
	HTML Basic
	HTML Styles
	HTML Forms and Input
3	W3Schools Online Web Tutorials
	DHTML Tutorial
	DHTML Introduction
	HTML Tutorial
	HTML Styles
4	W3Schools Online Web Tutorials
	HTML Tutorial
	HTML Frames
	HTML Styles
	HTML head Elements
5	W3Schools Online Web Tutorials
	DHTML Tutorial
	DHTML CSS
	HTML Tutorial
	HTML Basic
	HTML Frames

Table 5: Examples of Navigation History Sequences.

Table 6: 1-Length Page Support.

5.3   Experimental Data

This sub-section shows all kinds of data provided by the proposed system. An example of the general data provided by our system is given in Figure 10.

Figure10: An Example of the General Data.

Table 7: An Example of the Top Pages.

13
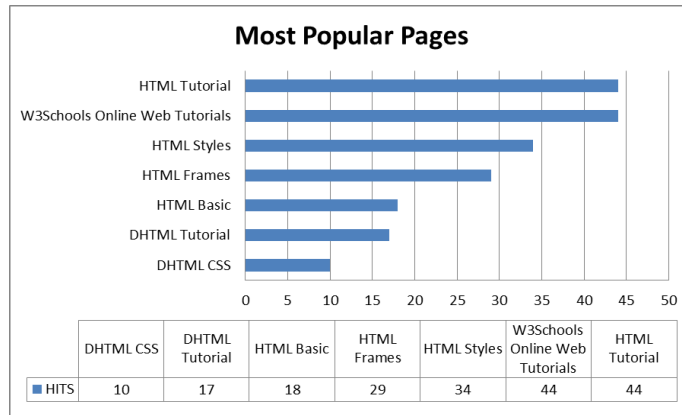
Figure 11: A Bar Chart for the Table 7.

Table 7 displays a list of the most popular Web pages visited in the Web site http://www.w3schools.com and the corresponding percentages. It shows how often the Web pages were viewed. Figure 11 graphically represents the most popular pages of the Table 7.

Top Entry Pages	
Pages	**Visits**
1 http://www.w3schools.com	44
Total	**44**

Table 8: Top Entry Pages.

Support for a Web Page		
Pages	**Relative Support**	**Absolute Support**
1 http://www.w3schools.com	94.7%	36
2 http://www.w3schools.com/html/default.asp	81.6%	31
3 http://www.w3schools.com/html/html_styles.asp	34.2%	13
4 http://www.w3schools.com/html/html_primary.asp	23.7%	9
5 http://www.w3schools.com/html/html_frames.asp	21.1%	8

Table 9: An Example of Page Supports.

Table 8 shows the top entry page, which is the page where users first enter the Website and Table 9 shows the supports of Web pages visited. A support gives how often the page was viewed. If the support is low, you may want to improve that Web page. On the other hand, if the support is high, you may make that page more accessible.

14

Table 10. An Example of Trails.

A trail is generated using the sequence of Web pages from a user during a session, ordered by time of access as shown in Table 10. From the navigation trails, you can return to upper or parent pages.

Figure 12: An Example of Visitor Information.

Figure 12 shows a list of visitors based on their IP addresses. This gives a rough estimation of the number of visitors using the system. Figure 13 displays the list of the most popular browsers used by visitors. This information helps in identifying what types of browsers and machines are used. It differentiates a mobile user from a desktop user.

Figure 13: An Example of Information about Browsers and Platforms Used.

15

5 Conclusion

The mobile Web recommender system proposed in this research is the process of analyzing the patterns of mobile users' online transactions and extracting knowledge about their Web access behavior. It is about extracting user's previously unknown, actionable intelligence from the Web logs. The Web usage mining system divides the Web usage mining process into different tasks. In usage data gathering, the proxy-based logging system collects Web usage data from Web logs. We used filtering techniques for usage data preparation to remove the undesired data from the Web logs to identify unique users and user sessions in order to prepare usage data. The usage patterns are discovered from the prepared data for navigation pattern analysis. Navigation patterns are then analyzed and discovered patterns are generated. The recommender system is based on these user navigation patterns. Finally, the usage patterns results are applied to mobile Web browsing.

The main objective of this research is to investigate a new method for enhancing mobile Web access using Web usage mining, which is a mixture of different technologies. Therefore, technologies such as data mining, database management, Web development, information retrieval, visualization and proxy construction have to work together to make Web usage mining a success. The Web usage mining technique is still immature. There are many challenges in building or deploying a recommender system. In order to make effective recommendations, a lot of data is required. The more data a recommender system has to work with, the stronger are the chances of getting good recommendations. Also, changing data and changing preferences are some of the important issues in building a recommender system.

Web usage mining is a new research field that is followed by scholars and commercial businesses, which utilizes data mining techniques to analyze the data stored in Web logs across the Internet. With, the growing number of mobile Web usage and lack of adequate resources for mobile Web user, the focus should be on building mobile friendly technology. The problems and opportunities that exist are plentiful. The future work will be to implement this approach on different types of smartphones, mobile devices, and wireless handheld gadgets. The future work may also include:

- XML is emerging as one of the key technologies in latest Web development. Migrate from traditional database-driven architecture to XML-driven one.
- Implement and strengthen the system on a large scale for mobile Web-browsing.
- With so much data, it is difficult to determine what is useful and how to display portion in a meaningful way. We need new methods of using visual display to gain insight into log data.

References

Agrawal, R & Srikant, R. (1995). Mining sequential patterns. In *Proceedings of the 11th International Conference on Data Engineering*, pages 3-14, Taipei, Taiwan.

Agrawal, R. & Srikant, R. (1994). Fast algorithms for mining association rules. In *Proceeding of the 20ᵗʰ Very Large DataBases Conference* (*VLDB*), pages 487-499, Santiago, Chile.

Clifton, C., Cooley, R., & Rennie, J. (2004). TopCat: data mining for topic identification in a text corpus. *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 8, pp. 949-964.

Cooley, R., Mobasher, B., & Srivastava, J. (1999). Data preparation for mining World Wide Web browsing patterns. *Knowledge and Information Systems*, vol. 1, pp. 5-32.

Eirinaki, M. & Vazirgiannis, M. (2003). Web mining for Web personalization. *Communications of ACM,* vol. 3, no. 1, pp. 1-27.

Hong, J. I. & Landay, J. A. (2001). WebQuilt: A framework for capturing and visualizing the Web experience. In *Proceedings of the 10ᵗʰ International World Wide Web Conference*, pages 717-724, Hong Kong.

Hu, W.-C., Zong, X., Chu, H.-J., & Chen, J.-F. (2002). Usage mining for the World Wide Web. In *Proceedings of the 6ᵗʰ World Multi-Conference on Systemics, Cybernetics and Informatics, SCI 2002*, pages 75-80, Orlando.

Perlfect Solutions. (n.d.). *CGI Environment Variables*. Retrieved May 15, 2010, from http://www.perlfect.com/articles/cgi_env.shtml

Resnick, P. & Varian, H. R. (1997). Recommender systems. *Communications of ACM,* vol. 40, no. 3, pp. 56-58.

Spiliopoulou, M. (2000). Web usage mining for site evaluation: Making a site better fit its users. *Communications of ACM*, vol. 43, no. 8, pp. 127-134.

W3C. (n.d.). *Common Log File Format*. Retrieved June 02, 2010, from http://www.w3.org/Daemon/User/Config/Logging.html

Xing, D. & Shen, J. (2004). Efficient data mining for Web navigation patterns. *Information and Software Technology*, vol. 46, no. 1, pp. 55-63.