Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data

Jaideep Srivastava* † , Robert Cooley‡ , Mukund Deshpande, Pang-Ning Tan
Department of Computer Science and Engineering
University of Minnesota
200 Union St SE
Minneapolis, MN 55455
{srivasta,cooley,deshpand,ptan}@cs.umn.edu

ABSTRACT

Web usage mining is the application of data mining techniques to discover usage patterns from Web data, in order to understand and better serve the needs of Web-based applications. Web usage mining consists of three phases, namely *preprocessing, pattern discovery,* and *pattern analysis.* This paper describes each of these phases in detail. Given its application potential, Web usage mining has seen a rapid increase in interest, from both the research and practice communities. This paper provides a detailed taxonomy of the work in this area, including research efforts as well as commercial offerings. An up-to-date survey of the existing work is also provided. Finally, a brief overview of the WebSIFT system as an example of a prototypical Web usage mining system is given.

Keywords: data mining, world wide web, web usage mining.

1. INTRODUCTION

The ease and speed with which business transactions can be carried out over the Web has been a key driving force in the rapid growth of electronic commerce. Specifically, e-commerce activity that involves the end user is undergoing a significant revolution. The ability to track users' browsing behavior down to individual mouse clicks has brought the vendor and end customer closer than ever before. It is now possible for a vendor to personalize his product message for individual customers at a massive scale, a phenomenon that is being referred to as *mass customization.*

The scenario described above is one of many possible applications of *Web Usage mining*, which is *the process of applying data mining techniques to the discovery of usage patterns from Web data*, targeted towards various applications. Data mining efforts associated with the Web, called *Web mining*, can be broadly divided into three classes, i.e. content mining, usage mining, and structure mining . Web Structure mining projects such as [34; 54] and Web Content mining projects such as [47; 21] are beyond the scope of this survey. An early taxonomy of Web mining is provided in [29], which also describes the architecture of the WebMiner system [42], one of the first systems for Web Usage mining. The proceedings of the recent WebKDD workshop [41], held in conjunction with the KDD-1999 conference, provides a sampling of some of the current research being performed in the area of Web Usage Analysis, including Web Usage mining.

This paper provides an up-to-date survey of Web Usage mining, including both academic and industrial research efforts, as well as commercial offerings. Section 2 describes the various kinds of Web data that can be useful for Web Usage mining. Section 3 discusses the challenges involved in discovering usage patterns from Web data. The three phases are *preprocessing, pattern discovery,* and *patterns analysis.* Section 4 provides a detailed taxonomy and survey of the existing efforts in Web Usage mining, and Section 5 gives an overview of the WebSIFT system [31], as a prototypical example of a Web Usage mining system. finally, Section 6 discusses privacy concerns and Section 7 concludes the paper.

*Can be contacted at jaideep@amazon.com

†Supported by NSF grant NSF/EIA-9818338

‡Supported by NSF grant EHR-9554517

2. WEB DATA

One of the key steps in Knowledge Discovery in Databases [33] is to create a suitable target data set for the data mining tasks. In Web Mining, data can be collected at the server-side, client-side, proxy servers, or obtained from an organization's database (which contains business data or consolidated Web data). Each type of data collection differs not only in terms of the location of the data source, but also the kinds of data available, the segment of population from which the data was collected, and its method of implementation.

There are many kinds of data that can be used in Web Mining. This paper classifies such data into the following types :

- **Content:** The *real* data in the Web pages, i.e. the data the Web page was designed to convey to the users. This usually consists of, but is not limited to, text and graphics.
- **Structure:** Data which describes the organization of the content. *Intra-page* structure information includes the arrangement of various HTML or XML tags within a given page. This can be represented as a tree structure, where the ⟨html⟩ tag becomes the root of the tree.

The principal kind of *inter-page* structure information is hyper-links connecting one page to another.

- **Usage:** Data that describes the pattern of usage of Web pages, such as IP addresses, page references, and the date and time of accesses.

- **User Profile:** Data that provides demographic information about users of the Web site. This includes registration data and customer profile information.

2.1 Data Sources

The usage data collected at the different sources will represent the navigation patterns of different segments of the overall Web traffic, ranging from single-user, single-site browsing behavior to multi-user, multi-site access patterns.

2.1.1 Server Level Collection

A Web server log is an important source for performing Web Usage Mining because it explicitly records the browsing behavior of site visitors. The data recorded in server logs reflects the (possibly concurrent) access of a Web site by multiple users. These log files can be stored in various formats such as Common log or Extended log formats. An example of Extended log format is given in Figure 2 (Section 3). However, the site usage data recorded by server logs may not be entirely reliable due to the presence of various levels of caching within the Web environment. Cached page views are not recorded in a server log. In addition, any important information passed through the POST method will not be available in a server log. Packet sniffing technology is an alternative method to collecting usage data through server logs. Packet sniffers monitor network traffic coming to a Web server and extract usage data directly from TCP/IP packets. The Web server can also store other kinds of usage information such as cookies and query data in separate logs. Cookies are tokens generated by the Web server for individual client browsers in order to automatically track the site visitors. Tracking of individual users is not an easy task due to the stateless connection model of the HTTP protocol. Cookies rely on implicit user cooperation and thus have raised growing concerns regarding user privacy, which will be discussed in Section 6. Query data is also typically generated by online visitors while searching for pages relevant to their information needs. Besides usage data, the server side also provides content data, structure information and Web page meta-information (such as the size of a file and its last modified time).

The Web server also relies on other utilities such as CGI scripts to handle data sent back from client browsers. Web servers implementing the CGI standard parse the URI[1] of the requested file to determine if it is an application program. The URI for CGI programs may contain additional parameter values to be passed to the CGI application. Once the CGI program has completed its execution, the Web server send the output of the CGI application back to the browser.

2.1.2 Client Level Collection

Client-side data collection can be implemented by using a remote agent (such as Javascripts or Java applets) or by modifying the source code of an existing browser (such as Mosaic or Mozilla) to enhance its data collection capabilities. The implementation of client-side data collection methods requires user cooperation, either in enabling the functionality of the Javascripts and Java applets, or to voluntarily use the modified browser. Client-side collection has an advantage over server-side collection because it ameliorates both the caching and session identification problems. However, Java applets perform no better than server logs in terms of determining the actual view time of a page. In fact, it may incur some additional overhead especially when the Java applet is loaded for the first time. Javascripts, on the other hand, consume little interpretation time but cannot capture all user clicks (such as reload or back buttons). These methods will collect only single-user, single-site browsing behavior. A modified browser is much more versatile and will allow data collection about a single user over multiple Web sites. The most difficult part of using this method is convincing the users to use the browser for their daily browsing activities. This can be done by offering incentives to users who are willing to use the browser, similar to the incentive programs offered by companies such as NetZero [9] and AllAdvantage [2] that reward users for clicking on banner advertisements while surfing the Web.

2.1.3 Proxy Level Collection

A Web proxy acts as an intermediate level of caching between client browsers and Web servers. Proxy caching can be used to reduce the loading time of a Web page experienced by users as well as the network traffic load at the server and client sides [27]. The performance of proxy caches depends on their ability to predict future page requests correctly. Proxy traces may reveal the actual HTTP requests from multiple clients to multiple Web servers. This may serve as a data source for characterizing the browsing behavior of a group of anonymous users sharing a common proxy server.

2.2 Data Abstractions

The information provided by the data sources described above can all be used to construct/identify several data abstractions, notably *users, server sessions, episodes, click-streams*, and *page views*. In order to provide some consistency in the way these terms are defined, the W3C Web Characterization Activity (WCA) [14] has published a draft of Web term definitions relevant to analyzing Web usage. A *user* is defined as a single individual that is accessing file from one or more Web servers through a browser. While this definition seems trivial, in practice it is very difficult to uniquely and repeatedly identify users. A user may access the Web through different machines, or use more than one agent on a single machine. A *page view* consists of every file that contributes to the display on a user's browser at one time. Page views are usually associated with a single user action (such as a mouse-click) and can consist of several files such as frames, graphics, and scripts. When discussing and analyzing user behaviors, it is really the aggregate page view that is of importance. The user does not explicitly ask for "n" frames and "m" graphics to be loaded into his or her browser, the user requests a "Web page." All of the information to determine which files constitute a page view is

[1]Uniform Resource Identifier (URI) is a more general definition that includes the commonly referred to Uniform Resource Locator (URL).

accessible from the Web server. A *click-stream* is a sequential series of page view requests. Again, the data available from the server side does not always provide enough information to reconstruct the full click-stream for a site. Any page view accessed through a client or proxy-level cache will not be "visible" from the server side. A *user session* is the click-stream of page views for a singe user across the entire Web. Typically, only the portion of each user session that is accessing a specific site can be used for analysis, since access information is not publicly available from the vast majority of Web servers. The set of page-views in a user session for a particular Web site is referred to as a *server session* (also commonly referred to as a *visit*). A set of server sessions is the necessary input for any Web Usage analysis or data mining tool. The end of a server session is defined as the point when the user's browsing session at that site has ended. Again, this is a simple concept that is very difficult to track reliably. Any semantically meaningful subset of a user or server session is referred to as an *episode* by the W3C WCA.

3. WEB USAGE MINING

As shown in Figure 1, there are three main tasks for performing Web Usage Mining or Web Usage Analysis. This section presents an overview of the tasks for each step and discusses the challenges involved.

3.1 Preprocessing

Preprocessing consists of converting the usage, content, and structure information contained in the various available data sources into the data abstractions necessary for pattern discovery.

3.1.1 Usage Preprocessing

Usage preprocessing is arguably the most difficult task in the Web Usage Mining process due to the incompleteness of the available data. Unless a client side tracking mechanism is used, only the IP address, agent, and server side click-stream are available to identify users and server sessions. Some of the typically encountered problems are:

- Single IP address/Multiple Server Sessions - Internet service providers (ISPs) typically have a pool of proxy servers that users access the Web through. A single proxy server may have several users accessing a Web site, potentially over the same time period.
- Multiple IP address/Single Server Session - Some ISPs or privacy tools randomly assign each request from a user to one of several IP addresses. In this case, a single server session can have multiple IP addresses.
- Multiple IP address/Single User - A user that accesses the Web from different machines will have a different IP address from session to session. This makes tracking repeat visits from the same user difficult.
- Multiple Agent/Singe User - Again, a user that uses more than one browser, even on the same machine, will appear as multiple users.

Assuming each user has now been identified (through cookies, logins, or IP/agent/path analysis), the click-stream for each user must be divided into sessions. Since page requests from other servers are not typically available, it is difficult to know when a user has left a Web site. A thirty minute timeout is often used as the default method of breaking a user's click-stream into sessions. The thirty minute timeout is based on the results of [23]. When a session ID is embedded in each URI, the definition of a session is set by the content server.

While the exact content served as a result of each user action is often available from the request field in the server logs, it is sometimes necessary to have access to the content server information as well. Since content servers can maintain state variables for each active session, the information necessary to determine exactly what content is served by a user request is not always available in the URI. The final problem encountered when preprocessing usage data is that of inferring cached page references. As discussed in Section 2.2, the only verifiable method of tracking cached page views is to monitor usage from the client side. The referrer field for each request can be used to detect some of the instances when cached pages have been viewed.

Figure 2 shows a sample log that illustrates several of the problems discussed above (The first column would not be present in an actual server log, and is for illustrative purposes only). IP address `123.456.78.9` is responsible for three server sessions, and IP addresses `209.456.78.2` and `209.45.78.3` are responsible for a fourth session. Using a combination of referrer and agent information, lines 1 through 11 can be divided into three sessions of `A-B-F-O-G`, `L-R`, and `A-B-C-J`. Path completion would add two page references to the first session `A-B-F-O-F-B-G`, and one reference to the third session `A-B-A-C-J`. Without using cookies, an embedded session ID, or a client-side data collection method, there is no method for determining that lines 12 and 13 are actually a single server session.

3.1.2 Content Preprocessing

Content preprocessing consists of converting the text, image, scripts, and other files such as multimedia into forms that are useful for the Web Usage Mining process. Often, this consists of performing content mining such as classification or clustering. While applying data mining to the content of Web sites is an interesting area of research in its own right, in the context of Web Usage Mining the content of a site can be used to filter the input to, or output from the pattern discovery algorithms. For example, results of a classification algorithm could be used to limit the discovered patterns to those containing page views about a certain subject or class of products. In addition to classifying or clustering page views based on topics, page views can also be classified according to their intended use [50; 30]. Page views can be intended to convey information (through text, graphics, or other multimedia), gather information from the user, allow navigation (through a list of hypertext links), or some combination uses. The intended use of a page view can also filter the sessions before or after pattern discovery. In order to run content mining algorithms on page views, the information must first be converted into a quantifiable format. Some version of the vector space model [51] is typically used to accomplish this. Text files can be broken up into vectors of words. Keywords or text descriptions can be substituted for graphics or multimedia. The content of static page views can be easily preprocessed by parsing the HTML and reformatting the information or running addi-

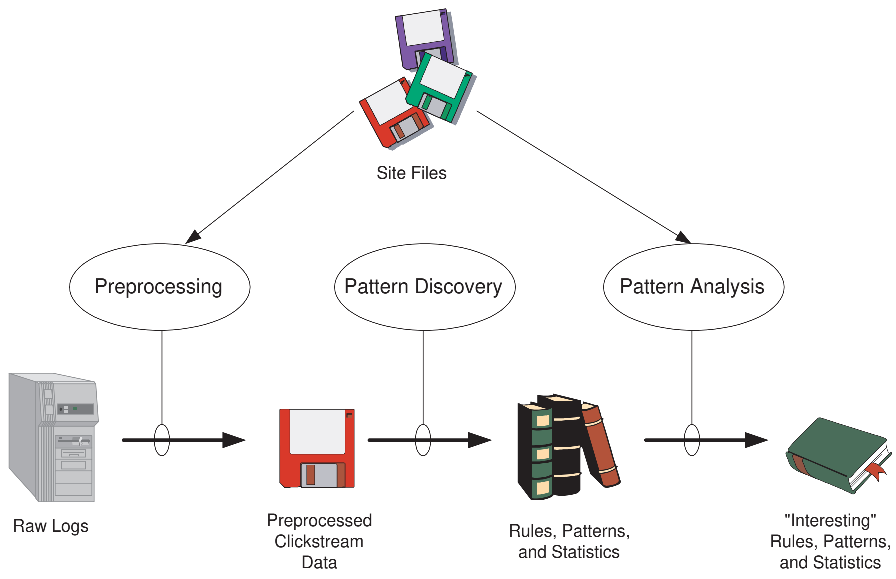

Figure 1: High Level *Web Usage Mining* Process

#	IP Address	Userid	Time	Method/ URL/ Protocol	Status	Size	Referrer	Agent
1	123.456.78.9	-	[25/Apr/1998:03:04:41 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.04 (Win95, I)
2	123.456.78.9	-	[25/Apr/1998:03:05:34 -0500]	"GET B.html HTTP/1.0"	200	2050	A.html	Mozilla/3.04 (Win95, I)
3	123.456.78.9	-	[25/Apr/1998:03:05:39 -0500]	"GET L.html HTTP/1.0"	200	4130	-	Mozilla/3.04 (Win95, I)
4	123.456.78.9	-	[25/Apr/1998:03:06:02 -0500]	"GET F.html HTTP/1.0"	200	5096	B.html	Mozilla/3.04 (Win95, I)
5	123.456.78.9	-	[25/Apr/1998:03:06:58 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
6	123.456.78.9	-	[25/Apr/1998:03:07:42 -0500]	"GET B.html HTTP/1.0"	200	2050	A.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
7	123.456.78.9	-	[25/Apr/1998:03:07:55 -0500]	"GET R.html HTTP/1.0"	200	8140	L.html	Mozilla/3.04 (Win95, I)
8	123.456.78.9	-	[25/Apr/1998:03:09:50 -0500]	"GET C.html HTTP/1.0"	200	1820	A.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
9	123.456.78.9	-	[25/Apr/1998:03:10:02 -0500]	"GET O.html HTTP/1.0"	200	2270	F.html	Mozilla/3.04 (Win95, I)
10	123.456.78.9	-	[25/Apr/1998:03:10:45 -0500]	"GET J.html HTTP/1.0"	200	9430	C.html	Mozilla/3.01 (X11, I, IRIX6.2, IP22)
11	123.456.78.9	-	[25/Apr/1998:03:12:23 -0500]	"GET G.html HTTP/1.0"	200	7220	B.html	Mozilla/3.04 (Win95, I)
12	209.456.78.2	-	[25/Apr/1998:05:05:22 -0500]	"GET A.html HTTP/1.0"	200	3290	-	Mozilla/3.04 (Win95, I)
13	209.456.78.3	-	[25/Apr/1998:05:06:03 -0500]	"GET D.html HTTP/1.0"	200	1680	A.html	Mozilla/3.04 (Win95, I)

Figure 2: Sample Web Server Log

tional algorithms as desired. Dynamic page views present more of a challenge. Content servers that employ personalization techniques and/or draw upon databases to construct the page views may be capable of forming more page views than can be practically preprocessed. A given set of server sessions may only access a fraction of the page views possible for a large dynamic site. Also the content may be revised on a regular basis. The content of each page view to be preprocessed must be "assembled", either by an HTTP request from a crawler, or a combination of template, script, and database accesses. If only the portion of page views that are accessed are preprocessed, the output of any classification or clustering algorithms may be skewed.

3.1.3 Structure Preprocessing

The structure of a site is created by the hypertext links between page views. The structure can be obtained and preprocessed in the same manner as the content of a site. Again, dynamic content (and therefore links) pose more problems than static page views. A different site structure may have to be constructed for each server session.

3.2 Pattern Discovery

Pattern discovery draws upon methods and algorithms developed from several fields such as statistics, data mining, machine learning and pattern recognition. However, it is not the intent of this paper to describe all the available algorithms and techniques derived from these fields. Interested readers should consult references such as [33; 24]. This section describes the kinds of mining activities that have been applied to the Web domain. Methods developed from other fields must take into consideration the different kinds of data abstractions and prior knowledge available for Web Mining. For example, in association rule discovery, the notion of a transaction for market-basket analysis does not take into consideration the order in which items are selected. However, in Web Usage Mining, a server session is an ordered sequence of pages requested by a user. Furthermore, due to the difficulty in identifying unique sessions, additional prior knowledge is required (such as imposing a default timeout period, as was pointed out in the previous section).

3.2.1 Statistical Analysis

Statistical techniques are the most common method to extract knowledge about visitors to a Web site. By analyzing the session file, one can perform different kinds of descriptive statistical analyses (frequency, mean, median, etc.) on variables such as page views, viewing time and length of a navigational path. Many Web traffic analysis tools produce a periodic report containing statistical information such as the most frequently accessed pages, average view time of a page or average length of a path through a site. This report may include limited low-level error analysis such as detecting unauthorized entry points or finding the most common invalid URI. Despite lacking in the depth of its analysis, this type of knowledge can be potentially useful for improving the system performance, enhancing the security of the system, facilitating the site modification task, and providing support for marketing decisions.

3.2.2 Association Rules

Association rule generation can be used to relate pages that are most often referenced together in a single server session. In the context of Web Usage Mining, association rules refer to sets of pages that are accessed together with a support value exceeding some specified threshold. These pages may not be directly connected to one another via hyperlinks. For example, association rule discovery using the Apriori algorithm [18] (or one of its variants) may reveal a correlation between users who visited a page containing electronic products to those who access a page about sporting equipment. Aside from being applicable for business and marketing applications, the presence or absence of such rules can help Web designers to restructure their Web site. The association rules may also serve as a heuristic for prefetching documents in order to reduce user-perceived latency when loading a page from a remote site.

3.2.3 Clustering

Clustering is a technique to group together a set of items having similar characteristics. In the Web Usage domain, there are two kinds of interesting clusters to be discovered : usage clusters and page clusters. Clustering of users tends to establish groups of users exhibiting similar browsing patterns. Such knowledge is especially useful for inferring user demographics in order to perform market segmentation in E-commerce applications or provide personalized Web content to the users. On the other hand, clustering of pages will discover groups of pages having related content. This information is useful for Internet search engines and Web assistance providers. In both applications, permanent or dynamic HTML pages can be created that suggest related hyperlinks to the user according to the user's query or past history of information needs.

3.2.4 Classification

Classification is the task of mapping a data item into one of several predefined classes [33]. In the Web domain, one is interested in developing a profile of users belonging to a particular class or category. This requires extraction and selection of features that best describe the properties of a given class or category. Classification can be done by using supervised inductive learning algorithms such as decision tree classifiers, naive Bayesian classifiers, k-nearest neighbor classifiers, Support Vector Machines etc. For example, classification on server logs may lead to the discovery of interesting rules such as : 30% of users who placed an online order in `/Product/Music` are in the 18-25 age group and live on the West Coast.

3.2.5 Sequential Patterns

The technique of sequential pattern discovery attempts to find inter-session patterns such that the presence of a set of items is followed by another item in a time-ordered set of sessions or episodes. By using this approach, Web marketers can predict future visit patterns which will be helpful in placing advertisements aimed at certain user groups. Other types of temporal analysis that can be performed on sequential patterns includes trend analysis, change point detection, or similarity analysis.

3.2.6 Dependency Modeling

Dependency modeling is another useful pattern discovery task in Web Mining. The goal here is to develop a model capable of representing significant dependencies among the various variables in the Web domain. As an example, one may be interested to build a model representing the different stages a visitor undergoes while shopping in an online store based on the actions chosen (ie. from a casual visitor to a serious potential buyer). There are several probabilistic learning techniques that can be employed to model the browsing behavior of users. Such techniques include Hidden Markov Models and Bayesian Belief Networks. Modeling of Web usage patterns will not only provide a theoretical framework for analyzing the behavior of users but is potentially useful for predicting future Web resource consumption. Such information may help develop strategies to increase the sales of products offered by the Web site or improve the navigational convenience of users.

3.3 Pattern Analysis

Pattern analysis is the last step in the overall Web Usage mining process as described in Figure 1. The motivation behind pattern analysis is to filter out uninteresting rules or patterns from the set found in the pattern discovery phase. The exact analysis methodology is usually governed by the application for which Web mining is done. The most common form of pattern analysis consists of a knowledge query mechanism such as SQL. Another method is to load usage data into a data cube in order to perform OLAP operations. Visualization techniques, such as graphing patterns or assigning colors to different values, can often highlight overall patterns or trends in the data. Content and structure information can be used to filter out patterns containing pages of a certain usage type, content type, or pages that match a certain hyperlink structure.

4. TAXONOMY AND PROJECT SURVEY

Since 1996 there have been several research projects and commercial products that have analyzed Web usage data for a number of different purposes. This section describes the dimensions and application areas that can be used to classify Web Usage Mining projects.

4.1 Taxonomy Dimensions

While the number of candidate dimensions that can be used to classify Web Usage Mining projects is many, there are five major dimensions that apply to every project - the data sources used to gather input, the types of input data, the number of users represented in each data set, the number of Web sites represented in each data set, and the application area focused on by the project. Usage data can either be gathered at the server level, proxy level, or client level, as discussed in Section 2.1. As shown in Figure 3, most projects make use of server side data. All projects analyze usage data and some also make use of content, structure, or profile data. The algorithms for a project can be designed to work on inputs representing one or many users and one or many Web sites. Single user projects are generally involved in the personalization application area. The projects that provide multi-site analysis use either client or proxy level input data in order to easily access usage data from more than one Web site. Most Web Usage Mining projects take single-site, multi-user, server-side usage data (Web server logs) as input.

4.2 Project Survey

As shown in Figures 3 and 4, usage patterns extracted from Web data have been applied to a wide range of applications. Projects such as [31; 55; 56; 58; 53] have focused on Web Usage Mining in general, without extensive tailoring of the process towards one of the various sub-categories. The WebSIFT project is discussed in more detail in the next section. Chen et al. [25] introduced the concept of maximal forward reference to characterize user episodes for the mining of traversal patterns. A maximal forward reference is the sequence of pages requested by a user up to the last page before backtracking occurs during a particular server session. The SpeedTracer project [56] from IBM Watson is built on the work originally reported in [25]. In addition to episode identification, SpeedTracer makes use of referrer and agent information in the preprocessing routines to identify users and server sessions in the absence of additional client side information. The Web Utilization Miner (WUM) system [55] provides a robust mining language in order to specify characteristics of discovered frequent paths that are interesting to the analyst. In their approach, individual navigation paths, called trails, are combined into an aggregated tree structure. Queries can be answered by mapping them into the intermediate nodes of the tree structure. Han et al. [58] have loaded Web server logs into a data cube structure in order to perform data mining as well as On-Line Analytical Processing (OLAP) activities such as roll-up and drill-down of the data. Their WebLogMiner system has been used to discover association rules, perform classification and time-series analysis (such as event sequence analysis, transition analysis and trend analysis). Shahabi et. al. [53; 59] have one of the few Web Usage mining systems that relies on client side data collection. The client side agent sends back page request and time information to the server every time a page containing the Java applet (either a new page or a previously cached page) is loaded or destroyed.

4.2.1 Personalization

Personalizing the Web experience for a user is the holy grail of many Web-based applications, e.g. individualized marketing for e-commerce [4]. Making dynamic recommendations to a Web user, based on her/his profile in addition to usage behavior is very attractive to many applications, e.g. *cross-sales* and *up-sales* in e-commerce. Web usage mining is an excellent approach for achieving this goal, as illustrated in [43] Existing recommendation systems, such as [8; 6], do not currently use data mining for recommendations, though there have been some recent proposals [16].

The WebWatcher [37], SiteHelper [45], Letizia [39], and clustering work by Mobasher et. al. [43] and Yan et. al. [57] have all concentrated on providing Web Site personalization based on usage information. Web server logs were used by Yan et. al. [57] to discover clusters of users having similar access patterns. The system proposed in [57] consists of an offline module that will perform cluster analysis and an online module which is responsible for dynamic link generation of Web pages. Every site user will be assigned to a single cluster based on their current traversal pattern. The links that are presented to a given user are dynamically selected based on what pages other users assigned to the same cluster have visited. The SiteHelper project learns a users preferences by looking at the page accesses for each user. A list of keywords from pages that a user has spent a significant amount of time viewing is compiled and presented to the user. Based on feedback about the keyword list, recommendations for other pages within the site are made. WebWatcher "follows" a user as he or she browses

Project	Application	Data	Source		Data	Type			User		Site	
	Focus	Server	Proxy	Client	Structure	Content	Usage	Profile	Single	Multi	Single	Multi
WebSIFT (CTS99)	General	x			x	x	x			x	x	
SpeedTracer (WYB98,CPY96)	General	x					x			x	x	
WUM (SF98)	General	x			x		x			x	x	
Shahabi (SZAS97,ZASS97)	General			x	x		x			x	x	
Site Helper (NW97)	Personalization	x				x	x		x		x	
Letizia (Lie95)	Personalization			x		x	x		x			x
Web Watcher (JFM97)	Personalization		x			x	x	x		x		x
Krishnapuram(NKJ99)	Personalization	x					x			x	x	
Analog (YJGD96)	Personalization	x					x			x	x	
Mobasher (MCS99)	Personalization	x			x		x			x	x	
Tuzhilin(PT98)	Business	x					x			x	x	
SurfAid	Business	x				x	x			x	x	
Buchner(BM98)	Business	x					x	x		x	x	
WebTrends,Hitlist,Accrue,etc.	Business	x					x			x	x	
WebLogMiner (ZXH98)	Business	x					x			x	x	
PageGather,SCML (PE98,PE99)	Site Modification	x			x	x	x			x	x	
Manley(Man97)	Characterization	x				x	x			x		x
Arlitt(AW96)	Characterization	x				x	x			x		x
Pitkow(PIT97,PIT98)	Characterization	x		x		x	x			x		x
Almeida(ABC96)	Characterization	x					x			x		x
Rexford(CKR98)	System Improve.	x	x				x			x	x	
Schechter(SKS98)	System Improve.	x					x			x	x	
Aggarwal(AY97)	System Improve.		x				x			x	x	

Figure 3: Web Usage Mining Research Projects and Products

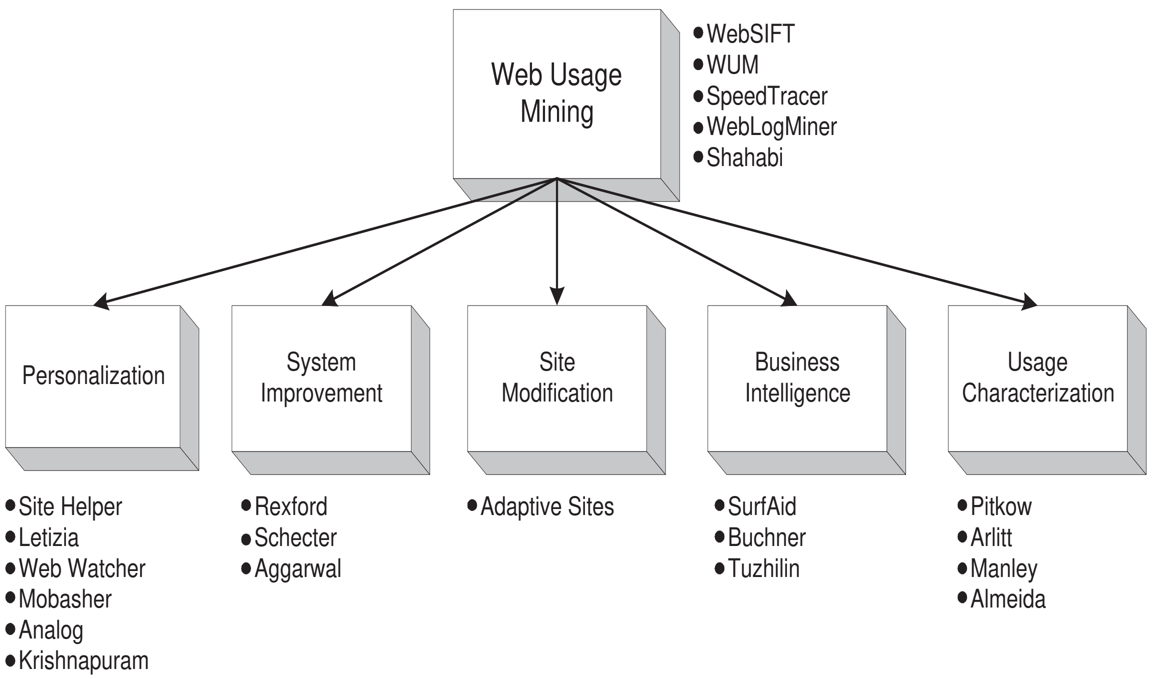

Figure 4: Major Application Areas for Web Usage Mining

the Web and identifies links that are potentially interesting to the user. The WebWatcher starts with a short description of a users interest. Each page request is routed through the WebWatcher proxy server in order to easily track the user session across multiple Web sites and mark any interesting links. WebWatcher learns based on the particular user's browsing plus the browsing of other users with similar interests. Letizia is a client side agent that searches the Web for pages similar to ones that the user has already viewed or bookmarked. The page recommendations in [43] are based on clusters of pages found from the server log for a site. The system recommends pages from clusters that most closely match the current session. Pages that have not been viewed and are not directly linked from the current page are recommended to the user. [44] attempts to cluster user sessions using a fuzzy clustering algorithm. [44] allows a page or user to be assigned to more than one cluster.

4.2.2 System Improvement

Performance and other service quality attributes are crucial to user satisfaction from services such as databases, networks, etc. Similar qualities are expected from the users of Web services. Web usage mining provides the key to understanding Web traffic behavior, which can in turn be used for developing policies for Web caching, network transmission [27], load balancing, or data distribution. Security is an acutely growing concern for Web-based services, especially as electronic commerce continues to grow at an exponential rate [32]. Web usage mining can also provide patterns which are useful for detecting intrusion, fraud, attempted break-ins, etc.

Almeida et al. [19] propose models for predicting the locality, both temporal as well as spatial, amongst Web pages requested from a particular user or a group of users accessing from the same proxy server. The locality measure can then be used for deciding pre-fetching and caching strategies for the proxy server. The increasing use of dynamic content has reduced the benefits of caching at both the client and server level. Schechter et. al. [52] have developed algorithms for creating path profiles from data contained in server logs. These profiles are then used to pre-generate dynamic HTML pages based on the current user profile in order to reduce latency due to page generation. Using proxy information from pre-fetching pages has also been studied by [27] and [17].

4.2.3 Site Modification

The attractiveness of a Web site, in terms of both content and structure, is crucial to many applications, e.g. a product catalog for e-commerce. Web usage mining provides detailed feedback on user behavior, providing the Web site designer information on which to base redesign decisions.

While the results of any of the projects could lead to redesigning the structure and content of a site, the adaptive Web site project (SCML algorithm) [48; 49] focuses on automatically changing the structure of a site based on usage patterns discovered from server logs. Clustering of pages is used to determine which pages should be directly linked.

4.2.4 Business Intelligence

Information on how customers are using a Web site is critical information for marketers of e-tailing businesses. Buchner et al [22] have presented a knowledge discovery process in order to discover marketing intelligence from Web data. They define a Web log data hypercube that will consolidate Web usage data along with marketing data for e-commerce applications. They identified four distinct steps in customer relationship life cycle that can be supported by their knowledge discovery techniques : customer attraction, customer retention, cross sales and customer departure. There are several commercial products, such as SurfAid [11], Accrue [1], NetGenesis [7], Aria [3], Hitlist [5], and WebTrends [13] that provide Web traffic analysis mainly for the purpose of gathering business intelligence. Accrue, NetGenesis, and Aria are designed to analyze e-commerce events such as products bought and advertisement click-through rates in addition to straight forward usage statistics. Accrue provides a path analysis visualization tool and IBM's SurfAid provides OLAP through a data cube and clustering of users in addition to page view statistics. Padmanabhan et. al. [46] use Web server logs to generate beliefs about the access patterns of Web pages at a given Web site. Algorithms for finding interesting rules based on the unexpectedness of the rule were also developed.

4.2.5 Usage Characterization

While most projects that work on characterizing the usage, content, and structure of the Web don't necessarily consider themselves to be engaged in data mining, there is a large amount of overlap between Web characterization research and Web Usage mining. Catledge et al. [23] discuss the results of a study conducted at the Georgia Institute of Technology, in which the Web browser Xmosaic was modified to log client side activity. The results collected provide detailed information about the user's interaction with the browser interface as well as the navigational strategy used to browse a particular site. The project also provides detailed statistics about occurrence of the various client side events such as the clicking the back/forward buttons, saving a file, adding to bookmarks etc. Pitkow et al. [36] propose a model which can be used to predict the probability distribution for various pages a user might visit on a given site. This model works by assigning a value to all the pages on a site based on various attributes of that page. The formulas and threshold values used in the model are derived from an extensive empirical study carried out on various browsing communities and their browsing patterns Arlitt et. al. [20] discuss various performance metrics for Web servers along with details about the relationship between each of these metrics for different workloads. Manley [40] develops a technique for generating a custom made benchmark for a given site based on its current workload. This benchmark, which he calls a *self-configuring benchmark*, can be used to perform scalability and load balancing studies on a Web server. Chi et. al. [35] describe a system called WEEV (Web Ecology and Evolution Visualization) which is a visualization tool to study the evolving relationship of web usage, content and site topology with respect to time.

5. WEBSIFT OVERVIEW

The WebSIFT system [31] is designed to perform Web Usage Mining from server logs in the extended NSCA format (includes referrer and agent fields). The preprocessing algorithms include identifying users, server sessions, and inferring cached page references through the use of the referrer field. The details of the algorithms used for these steps are contained in [30]. In addition to creating a server session

file, the WebSIFT system performs content and structure preprocessing, and provides the option to convert server sessions into episodes. Each episode is either the subset of all content pages in a server session, or all of the navigation pages up to and including each content page. Several algorithms for identifying episodes (referred to as transactions in the paper) are described and evaluated in [28].
The server session or episode files can be run through sequential pattern analysis, association rule discovery, clustering, or general statistics algorithms, as shown in Figure 5. The results of the various knowledge discovery tools can be analyzed through a simple knowledge query mechanism, a visualization tool (association rule map with confidence and support weighted edges), or the information filter (OLAP tools such as a data cube are possible as shown in Figure 5, but are not currently implemented). The information filter makes use of the preprocessed content and structure information to automatically filter the results of the knowledge discovery algorithms for patterns that are potentially interesting. For example, usage clusters that contain page views from multiple content clusters are potentially interesting, whereas usage clusters that match content clusters may not be interesting. The details of the method the information filter uses to combine and compare evidence from the different data sources are contained in [31].

6. PRIVACY ISSUES

Privacy is a sensitive topic which has been attracting a lot of attention recently due to rapid growth of e-commerce. It is further complicated by the global and self-regulatory nature of the Web. The issue of privacy revolves around the fact that most users want to maintain strict anonymity on the Web. They are extremely averse to the idea that someone is monitoring the Web sites they visit and the time they spend on those sites.
On the other hand, site administrators are interested in finding out the demographics of users as well as the usage statistics of different sections of their Web site. This information would allow them to improve the design of the Web site and would ensure that the content caters to the largest population of users visiting their site. The site administrators also want the ability to identify a user uniquely every time she visits the site, in order to personalize the Web site and improve the browsing experience.
The main challenge is to come up with guidelines and rules such that site administrators can perform various analyses on the usage data without compromising the identity of an individual user. Furthermore, there should be strict regulations to prevent the usage data from being exchanged/sold to other sites. The users should be made aware of the privacy policies followed by any given site, so that they can make an informed decision about revealing their personal data. The success of any such guidelines can only be guaranteed if they are backed up by a legal framework.
The W3C has an ongoing initiative called Platform for Privacy Preferences (P3P) [10; 38]. P3P provides a protocol which allows the site administrators to publish the privacy policies followed by a site in a machine readable format. When the user visits the site for the first time the browser reads the privacy policies followed by the site and then compares that with that security setting configured by the user. If the policies are satisfactory the browser continues requesting pages from the site, otherwise a negotiation protocol is used to arrive at a setting which is acceptable to the user. Another aim of P3P is to provide guidelines for independent organizations which can ensure that sites comply with the policy statement they are publishing [12].
The European Union has taken a lead in setting up a regulatory framework for Internet Privacy and has issued a directive which sets guidelines for processing and transfer of personal data [15]. Unfortunately in U.S. there is no unifying framework in place, though U.S. Federal Trade Commission (FTC) after a study of commercial Web sites has recommended that Congress develop legislation to regulate the personal information being collected at Web sites[26].

7. CONCLUSIONS

This paper has attempted to provide an up-to-date survey of the rapidly growing area of Web Usage mining. With the growth of Web-based applications, specifically electronic commerce, there is significant interest in analyzing Web usage data to better understand Web usage, and apply the knowledge to better serve users. This has led to a number of commercial offerings for doing such analysis. However, Web Usage mining raises some hard scientific questions that must be answered before robust tools can be developed. This article has aimed at describing such challenges, and the hope is that the research community will take up the challenge of addressing them.

8. REFERENCES

[1] Accrue. http://www.accrue.com.

[2] Alladvantage. http://www.alladvantage.com.

[3] Andromedia aria. http://www.andromedia.com.

[4] Broadvision. http://www.broadvision.com.

[5] Hit list commerce. http://www.marketwave.com.

[6] Likeminds. http://www.andromedia.com.

[7] Netgenesis. http://www.netgenesis.com.

[8] Netperceptions. http://www.netperceptions.com.

[9] Netzero. http://www.netzero.com.

[10] Platform for privacy project. http://www.w3.org/P3P/.

[11] Surfaid analytics. http://surfaid.dfw.ibm.com.

[12] Truste: Building a web you can believe in. http://www.truste.org/.

[13] Webtrends log analyzer. http://www.webtrends.com.

[14] World wide web committee web usage characterization activity. http://www.w3.org/WCA.

[15] European commission. the directive on the protection of individuals with regard ot the processing of personal data and on the free movement of such data. http://www2.echo.lu/, 1998.

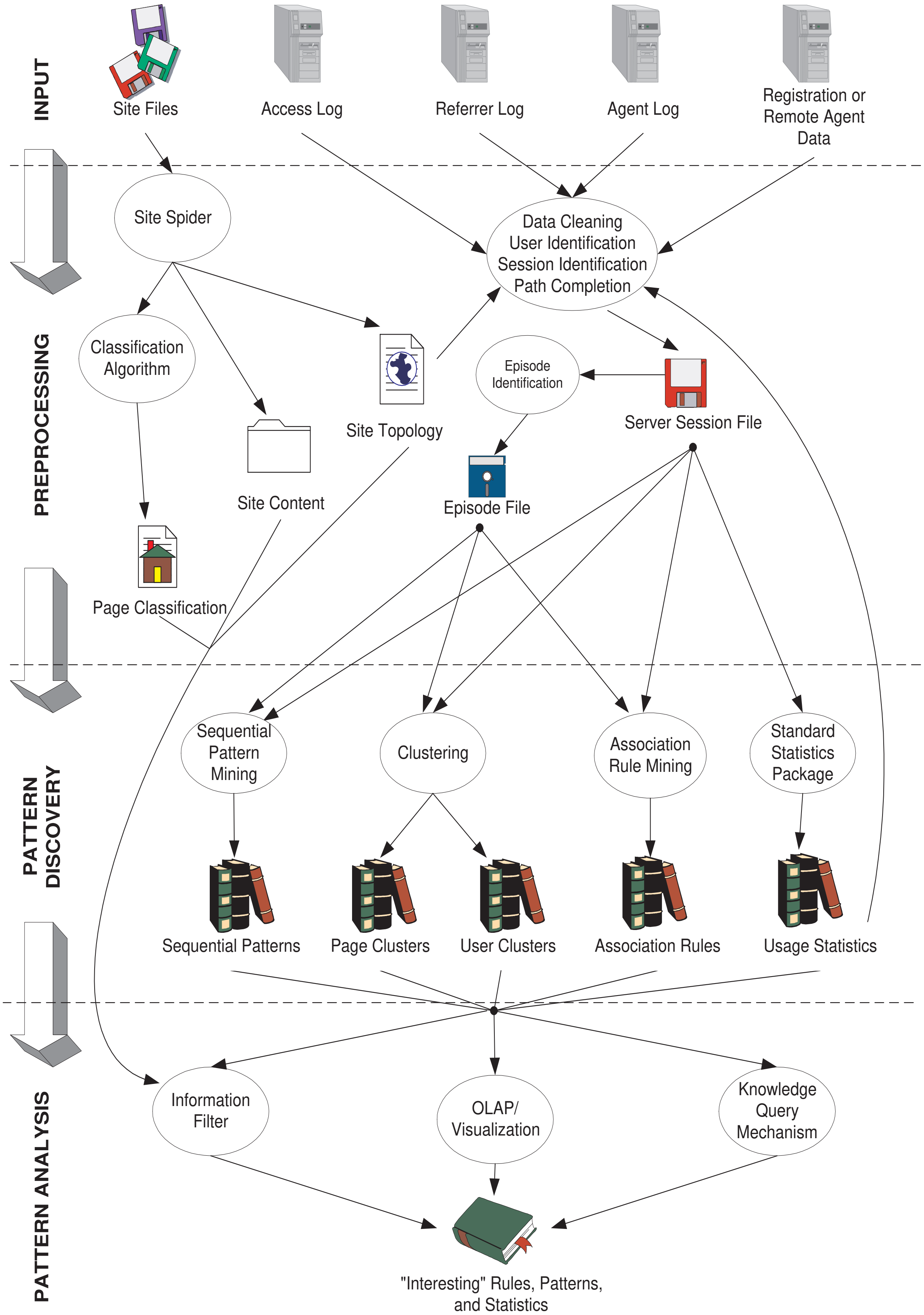

Figure 5: Architecture for the WebSIFT System

[16] Data mining: Crossing the chasm, 1999. Invited talk at the 5th ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining(KDD99).

[17] Charu C Aggarwal and Philip S Yu. On disk caching of web objects in proxy servers. In *CIKM 97*, pages 238–245, Las Vegas, Nevada, 1997.

[18] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. of the 20th VLDB Conference*, pages 487–499, Santiago, Chile, 1994.

[19] Virgilio Almeida, Azer Bestavros, Mark Crovella, and Adriana de Oliveira. Characterizing reference locality in the www. Technical Report TR-96-11, Boston University, 1996.

[20] Martin F Arlitt and Carey L Williamson. Internet web servers: Workload characterization and performance implications. *IEEE/ACM Transactions on Networking*, 5(5):631–645, 1997.

[21] M. Balabanovic and Y. Shoham. Learning information retrieval agents: Experiments with automated web browsing. In *On-line Working Notes of the AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments*, 1995.

[22] Alex Buchner and Maurice D Mulvenna. Discovering internet marketing intelligence through online analytical web usage mining. *SIGMOD Record*, 27(4):54–61, 1998.

[23] L. Catledge and J. Pitkow. Characterizing browsing behaviors on the world wide web. *Computer Networks and ISDN Systems*, 27(6), 1995.

[24] M.S. Chen, J. Han, and P.S. Yu. Data mining: An overview from a database perspective. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):866–883, 1996.

[25] M.S. Chen, J.S. Park, and P.S. Yu. Data mining for path traversal patterns in a web environment. In *16th International Conference on Distributed Computing Systems*, pages 385–392, 1996.

[26] Roger Clarke. Internet privacy concerns conf the case for intervention. 42(2):60–67, 1999.

[27] E. Cohen, B. Krishnamurthy, and J. Rexford. Improving end-to-end performance of the web using server volumes and proxy filters. In *Proc. ACM SIGCOMM*, pages 241–253, 1998.

[28] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Grouping web page references into transactions for mining world wide web browsing patterns. In *Knowledge and Data Engineering Workshop*, pages 2–9, Newport Beach, CA, 1997. IEEE.

[29] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Web mining: Information and pattern discovery on the world wide web. In *International Conference on Tools with Artificial Intelligence*, pages 558–567, Newport Beach, 1997. IEEE.

[30] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1(1), 1999.

[31] Robert Cooley, Pang-Ning Tan, and Jaideep Srivastava. Discovery of interesting usage patterns from web data. Technical Report TR 99-022, University of Minnesota, 1999.

[32] T. Fawcett and F. Provost. Activity monitoring: Noticing interesting changes in behavior. In *Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 53–62, San Diego, CA, 1999. ACM.

[33] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery: An overview. In *Proc. ACM KDD*, 1994.

[34] David Gibson, Jon Kleinberg, and Prabhakar Raghavan. Inferring web communities from link topology. In *Conference on Hypertext and Hypermedia*. ACM, 1998.

[35] Chi E. H., Pitkow J., Mackinlay J., Pirolli P., Gossweiler, and Card S. K. Visualizing the evolution of web ecologies. In *CHI '98*, Los Angeles, California, 1998.

[36] Bernardo Huberman, Peter Pirolli, James Pitkow, and Rajan Kukose. Strong regularities in world wide web surfing. Technical report, Xerox PARC, 1998.

[37] T. Joachims, D. Freitag, and T. Mitchell. Webwatcher: A tour guide for the world wide web. In *The 15th International Conference on Artificial Intelligence*, Nagoya, Japan, 1997.

[38] Reagle Joseph and Cranor Lorrie Faith. The platform for privacy preferences. 42(2):48–55, 1999.

[39] H. Lieberman. Letizia: An agent that assists web browsing. In *Proc. of the 1995 International Joint Conference on Artificial Intelligence*, Montreal, Canada, 1995.

[40] Stephen Lee Manley. *An Analysis of Issues Facing World Wide Web Servers*. Undergraduate, Harvard, 1997.

[41] B. Masand and M. Spiliopoulou, editors. *Workshop on Web Usage Analysis and User Profiling (WebKDD)*, 1999.

[42] B. Mobasher, N. Jain, E. Han, and J. Srivastava. Web mining: Pattern discovery from world wide web transactions. (TR 96-050), 1996.

[43] Bamshad Mobasher, Robert Cooley, and Jaideep Srivastava. Creating adaptive web sites through usage-based clustering of urls. In *Knowledge and Data Engineering Workshop*, 1999.

[44] Olfa Nasraoui, Raghu Krishnapuram, and Anupam Joshi. Mining web access logs using a fuzzy relational clustering algorithm based on a robust estimator. In *Eighth International World Wide Web Conference*, Toronto, Canada, 1999.

[45] D.S.W. Ngu and X. Wu. Sitehelper: A localized agent that helps incremental exploration of the world wide web. In *6th International World Wide Web Conference*, Santa Clara, CA, 1997.

[46] Balaji Padmanabhan and Alexander Tuzhilin. A belief-driven method for discovering unexpected patterns. In *Fourth International Conference on Knowledge Discovery and Data Mining*, pages 94–100, New York, New York, 1998.

[47] M. Pazzani, L. Nguyen, and S. Mantik. Learning from hotlists and coldlists: Towards a www information filtering and seeking agent. In *IEEE 1995 International Conference on Tools with Artificial Intelligence*, 1995.

[48] Mike Perkowitz and Oren Etzioni. Adaptive web sites: Automatically synthesizing web pages. In *Fifteenth National Conference on Artificial Intelligence*, Madison, WI, 1998.

[49] Mike Perkowitz and Oren Etzioni. Adaptive web sites: Conceptual cluster mining. In *Sixteenth International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 1999.

[50] Peter Pirolli, James Pitkow, and Ramana Rao. Silk from a sow's ear: Extracting usable structures from the web. In *CHI-96*, Vancouver, 1996.

[51] G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.

[52] S. Schechter, M. Krishnan, and M. D. Smith. Using path profiles to predict http requests. In *7th International World Wide Web Conference*, Brisbane, Australia, 1998.

[53] Cyrus Shahabi, Amir M Zarkesh, Jafar Adibi, and Vishal Shah. Knowledge discovery from users web-page navigation. In *Workshop on Research Issues in Data Engineering*, Birmingham, England, 1997.

[54] E. Spertus. Parasite : Mining structural information on the web. *Computer Networks and ISDN Systems: The International Journal of Computer and Telecommunication Networking*, 29:1205–1215, 1997.

[55] Myra Spiliopoulou and Lukas C Faulstich. Wum: A web utilization miner. In *EDBT Workshop WebDB98*, Valencia, Spain, 1998. Springer Verlag.

[56] Kun-lung Wu, Philip S Yu, and Allen Ballman. Speedtracer: A web usage mining and analysis tool. *IBM Systems Journal*, 37(1), 1998.

[57] T. Yan, M. Jacobsen, H. Garcia-Molina, and U. Dayal. From user access patterns to dynamic hypertext linking. In *Fifth International World Wide Web Conference*, Paris, France, 1996.

[58] O. R. Zaiane, M. Xin, and J. Han. Discovering web access patterns and trends by applying olap and data mining technology on web logs. In *Advances in Digital Libraries*, pages 19–29, Santa Barbara, CA, 1998.

[59] Amir Zarkesh, Jafar Adibi, Cyrus Shahabi, Reza Sadri, and Vishal Shah. Analysis and design of server informative www-sites. In *Sixth International Conference on Information and Knowledge Management*, Las Vegas, Nevada, 1997.

About the Authors :

Jaideep Srivastava received the B.Tech. degree in computer science from the Indian Institute of Technology, Kanpur, India, in 1983, and the M.S. and Ph.D. degrees in computer science from the University of California, Berkeley, in 1985 and 1988, respectively. Since 1988 he has been on the faculty of the Computer Science Department, University of Minnesota, Minneapolis, where he is currently an Associate Professor. In 1983 he was a research engineer with Uptron Digital Systems, Lucknow, India. He has published over 110 papers in refereed journals and conferences in the areas of databases, parallel processing, artificial intelligence, and multi-media. His current research is in the areas of databases, distributed systems, and multi-media computing. He has given a number of invited talks and participated in panel discussions on these topics. Dr. Srivastava is a senior member of the IEEE Computer Society and the ACM. His professional activities have included being on various program committees, and refereeing for journals, conferences, and the NSF.

Robert Cooley is currently pursuing a Ph.D. in computer science at the University of Minnesota. He received an M.S. in computer science from Minnesota in 1998. His research interests include Data Mining and Information Retrieval.

Mukund Deshpande is a Ph.D. student in the Department of Computer Science at the University of Minnesota. He received an M.E. in system science & automation from Indian Institute of Science, Bangalore, India in 1997.

Pang-Ning Tan is currently working towards his Ph.D. in Computer Science at University of Minnesota. His primary research interest is in Data Mining. He received an M.S. in Physics from University of Minnesota in 1996.