**WEB SERVER TO CLIENT COMMUNICATION FOR WEB USAGE DATA ANALYSIS**

**By: Antoneta Skënderi**

**Table of contents**

**Abstract…………………………………………………………………………..…………….5**

Introduction**……..………………………………………………..…………..………..………..6**

Methodology**………………………….……………………………..…………………...….….6**

**CHAPTER 1**

**Data Preprocessing**

1.1.Customer Behavior Prediction Using Web Usage Mining**………………………..……………7**

1.2.PC Configuration Retrieval System on Online Server**…………………….……..……………8**

1.3.Web Server Log Analysis System**……………………..…………………………….…………9**

1.4.Network Based Stock Price System**……………………………………………...……………9**

1.5.Mining Document Structure**……………………………………………………...…………..10**

**CHAPTER 2**

**Web Usage Mining**

2.1.Web Content Mining**……………………..…………………………………………………..12**

2.2.Web Structured Mining**…………………………………………………………..…………..13**

2.3.Mining the Web nowadays**……………………………………………………….…………..14**

2.4.Server-Side Techniques**…………………………………………………………..…………..15**

2.5.Cookies**……………………..………………………………………………………………..16**

**CHAPTER 3**

**Improving Workflow Management Support with Client-side Monitoring**

3.1.Web Usage Mining**……………………..…………………………………………………..18**

3.2.Data collection**……………………..……………………………………………...………..19**

3.3.Data preprocessing**……………………………………………………………...…………..20**

3.4.Data Cleaning**……………………..…………………………………………………….…..20**

3.5.User Identification**……………………..………………………………………………..…..20**

3.6.Session Identification**……………………..…………………………………………..……..21**

3.7.Path Completion**……………………..………………………………………………..……..23**

**CHAPTER 4**

**Data analysis part**

Analysis of Web server log by Web usage mining for extracting users patterns

4.1.Pattern discovery and analysis**……………………..………………………………………..25**

**CHAPTER 5**

**Implementation part**

Contents of Log File

5.1.NCSA Log Formats**……………………..………………………………………………..…..27**

5.2.NCSA Separate Log Format**……………………..……………………………………….…..28**

5.3.W3C Extended Log Format**……………………..………………………………………..…..28**

5.4.Microsoft IIS Log File**……………………..……………………………………………..…..29**

5.5.Sun One Web Server**……………………..…………………………………………………...30**

5.6.Request & Response by the Server**…………………………………………...…..…………..30**

5.7.Connection Errors**……………………..……………………………………………………..31**

5.8. Experimental results**……………………..…………………………………………………..34**

**CHAPTER 6**

**Server-Side Programming to Augment Basic Server-Side Data**

6.1.“User-Centric” Surf Mining**……………………..…………………………………….……..39**

6.2.Generic Agents to help Knowledge Work**………………………………………..…………..40**

6.3.Improving Personalization and Recommender Systems**……………..…………..…………..40**

6.4.Client-Side Monitoring Today**……………………..……………………………………..…..40**

6.5.Goals for Client-Side Monitoring**………………………………………………...…………..41**

6.6.Framework for User Monitoring**……………………………………………..…..…………..42**

6.7.Monitoring Internet Explorer**……………………..………………………………...………..43**

6.8.Link mining**……………………..……………………………………………………..……..45**

6.9.Monitoring Microsoft Word**………………………………………………….…..…………..46**

6.10.Compositional analysis**………………………………………………………....…………..46**

6.11.Monitoring Microsoft Excel**……………………………………………...……..…………..47**

6.12.Windows clipboard: The link among applications**……………………….……..…………..47**

6.13.Web usage mining applications**……………………..………………………………..……..48**

**Conclusions…………………………………...……………………………………..…………..49**

**Future Development………………………………………...…………………………………..50**

**References………………………………………………………………....…………………….51**

**Abstract**

We usually come across areas where online data is presented as very important to the business as well as security analysis. E-commerce as well as website usage data are of major importance these days. Our proposed project introduces a system that allows an Internet server to record Internet data analytics of Internet usage patterns of various Internet connections. The system is configured to record clicks on web pages, SEO score calculated according to a search engine ranking algorithm, as well as other parameters associated with each Web page.

The server has an integrated configuration logic required to scan live crawlers-like web pages and then record / save that data for later processing. Since data can be large and great computing power is required to maintain the process of this data. This calculation can lead to server overload thus hindering its further indexing performance. So we propose a client application that communicates with the Web server to retrieve this data and allows processing and manipulation based on these recorded Web service data. This powerful system allows direct scanning and recording of Web analytics data as well as their processing by communicating this data to another powerful application in order to avoid getting the processing load from the server.

**Strengths**

• The system is easy to install.

• Will help the website owner to organize the data in a more useful way.

• Will also help improve the website by studying data, what its customers like most about its website.

• It is cost-effective.

• Maximizes accuracy.

**Disadvantages**

• Uses a lot of memory on the server to store data.

# **Introduction**

The WWW continues to grow at an astonishing rate in both information and user perspective. The rate of information on the Internet is growing at an understandable rate, similar to the mystical size of planets and stars. The Internet has become a place where a massive amount of information and data is being generated every day. Over the past decade with the ever-increasing use of the WWW, Web mining has established itself as an important area of ​​research.

Web mining is used to analyze users using the WWW who leave ample information in the Web registry, which is structurally complex and of an additional nature. A Log file is a record that records everything that goes in and out of a particular server. Analyzing such data will bring knowledge, but pre-processing of this data is required before analyzing them. After analyzing the Log File, they provide activities to users for a potentially long period of time. They can be collected from the Web server, Proxy server and Web client. These records, when properly mined, provide useful information for decision making. They contain information such as username, IP address, time stamp, transferred bytes, referenced URL, User Agent.

Internet mining facilities are large, heterogeneous and distribute documents. The logistics structure of the Web is a graph structured by documents and hyperlinks, mining results perhaps in Web content or Web structures. Online mining is divided into three types. They are the Internet content mine, the Web structure mine and the Web usage mine.

Web Content Mining deals with the discovery of useful information from online content or data or documents or services.

Web Structure Mining undermines the structure of hyperlinks within the network itself. The structure represents the graph of the link to a site or between pages.

Web Usage Mining undermines the registry data stored on the Web server.

**Methodology**

There is a growing trend among enterprises, organizations and individuals to gather information from registry files to gather information about users, but it is a challenging task for them to meet the needs of the user. Web mines have valuable uses for business marketing and a direct impact on the success of their promotional strategies and web traffic. This information is collected daily and continues to be constantly analyzed. Analyzing this relevant information will help companies develop the most effective promotions, Internet access, communication and inter-enterprise structure and productive marketing skills through the mining of website usage server file server and its format. Next we have extracted the user models using groups, decision trees and If-Then rules.

Extended work for this research is extracting user clicks based on the log file. This should have an in-depth overview to record files that save user clicks.

**CHAPTER 1**

**Data Preprocessing**

The Log file contains intangible attributes. So before extracting the Log file, pre-processing must be done. The general preprocessing techniques applied to the data are Purification, Integration, Transformation and Reduction. By applying the incomplete attributes of the above pre-processing techniques, noisy data which contains errors and inconsistent data that have inconsistencies can be removed. By applying data preprocessing we improve data quality.

Once the cleared data is transformed, user sessions can be tracked to identify the user, and user models can be extracted from it.

# **1.1.Customer Behavior Prediction Using Web Usage Mining**

Mine use of the Web to pre-record customer behavior and flow on a website and then minimize through the use of data on patterns and behaviors. This issue is a part of the world of E-commerce that allows websites to go through traffic tasks previously registered online. E-commerce sites analyze this data in order to provide better performance and others suggest better products and services for customers by identifying them next time.

The system is tuned to record patterns and online purchases / purchases and track others to analyze who tends to make forecast statistics. The system scans for tracking user budgets, calculating previous years, user bounce rate - number of users returning from page by page and other page usage factors. Random user factors allow site owners to make the right changes and give customers exactly what they need. This allows more customer purchases and thus more utilization. E-commerce sites need to be checked and registered in order to be pre-registered to check their performance and in order to be constantly optimized according to customer needs.

**Strengths**

• The system is easy to install.

• Will help the owner take the user's next step.

• He can develop his future plans based on this study.

**Disadvantages**

• Uses a lot of memory on the server to store data.

• It is not so accurate.

# **1.2.PC Configuration Retrieval System on Online Server**

Various software / game applications require precise PC configurations to run on them. Many of the PC users are non-technical people who are not aware of their correct PC configuration. So for software companies it becomes inefficient to manually interact with each client and thus suggest PC configurations and suggest the changes they need to make to run a particular application on their PC. Well here we propose a solution to this issue aiming at a project of suggesting retriever configuration and PC configuration.

First, the system should be guided with different programs / game applications, minimal configuration suggestions. So now whenever a user wants to know if a software / game will work properly on his PC he does not need to check it manually. The company website will consist of this application which can be downloaded and run automatically without installation. Now asks the user about the software / game whose compatibility he should check with his PC.

The application now scans the user's PC for various PC configuration data and suggests what the PC update user needs to do to run that particular system on his PC. If the configuration meets the needs, it suggests that his computer be fit for the application. The software also sends this username data to the company so that the company stays in touch with the user later.

**Strengths**

• The system is easy to install.

• Will help the user know whether his PC is compatible with the program or not.

• It saves the user if the program is not compatible with his PC.

• This system will send information to the server for user configuration.

**Disadvantages**

• The user must download and install it, to check if the configuration of his PC is compatible with the program or not.

**1.3.Web Server Log Analysis System**

Web usage data is of great importance these days. Websites used every day and different users who enter a website are two main categories of data of great importance. Here we propose an online mining algorithm that proves better than most traditional online mining algorithms. Here we track data online and use the E-web miner algorithm to analyze and report online logs.

The algorithm works on E-commerce data by constantly scanning and going through the Internet registry looking for patterns as suggested by user conditions. The algorithm was created to look for different models that appear to be in any logical order. It is built to provide analytical data according to predefined algorithms built for maximum performance and minimum processing time.

# **Strengths**

# • The system is easy to install.

# • Will help the website owner to organize the data in a more useful way.

# • Will also help improve the website by studying data, what his clients like most on his website.

# • It is cost-effective.

# • Maximizes accuracy.

# **Disadvantages**

# • Uses a lot of memory on the server to store data.

# **1.4.Network Based Stock Price System**

We usually encounter systems that require an Internet server to constantly keep up to date with price updates. Many applications such as stock market client applications need to constantly interact with their server in order to update those prices from time to time. The system must not only update the price, but must perform this task accurately and quickly. As stock prices continue to change every moment, it becomes possible to have good communication between server and client applications. We propose here an HTTP-based class for client-server communication for stock price updates. The system tends to provide a powerful stock price updating system. The client system uses continuous refresh functions to update the stock price every second.

**The system works as follows:**

• Server allows the user to update the stock price when it is needed or a random simulator can be used.

• Further, the client application now needs to be constantly connected to the server for updates.

• The client application uses HTTP server requests to change parameters every second.

• The server returns the HTTP response settings with updated values as soon as an update occurs.

• Now the client system is constantly updating its screen in order to update new values.

• The client algorithm constantly compares the value received with the previous one.

• Updates found are reflected in the client application.

**Strengths**

• The rating is updated in miles-seconds.

• The user can look at their stock rate and make a quick decision.

**Disadvantages**

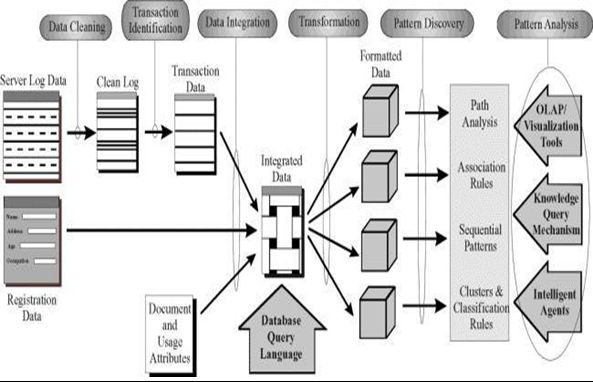
• The user must be constantly connected to the Internet.

## 1.5.Mining Document Structure

Analyzes tree-like structure of the page to describe the uses of HTML, XML tags. Web Content Mining in Figure 1 is mining, modeling, and extracting knowledge from Web content.

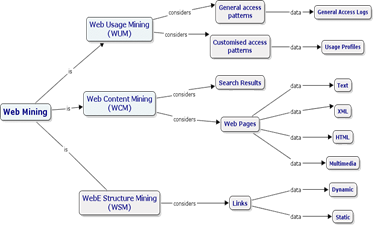
**They are of 2 types:**

* Information retrieval view
* Database View



## Figure 1: Data Pre-Processing

The data obtained now are clean and can be analyzed. The online extraction process is considered for the analysis of the pre-processed Log File.

***Figure 2***: Web Mining

**CHAPTER 2**

**Web Usage Mining**

The purpose of using the Web is to log in to the server registers (log files) that store transactions performed on the Web in order to find patterns that reveal the use of clients. We can also distinguish here:

• General follow-up of the input model. Here we combine the login patterns of a group rather than an individual to take a trend that allows us to organize the structure of the Web in such a way that the user is comfortable.

• Tracing the personalized access model. Here we collect information about a client’s behavior with the website. Based on the information gathered suggestions and tips are given to improve the quality.

## 2.1.Web Content Mining

Here information is collected about content research to identify user models. There are two main types of Web Content Mining strategies:

***View of information retrieval:*** R. Kosala et al. summarized the research work done on unstructured data and semi-structured data from the point of view of information retrieval. Research has found that research uses frequent words, which is usually a single word. These single words are considered as training corpses for ranking content on the Web based on the number of referrals. For semi-structured data, all works utilize HTML structures within documents and some used the hyperlink structure between documents to represent documents.

**Database View**

In terms of database layout, in order to have a better management and query of online information, the mine always tries to derive the completion of the website structure to transform a website to be made a database.

## 2.2.Web Structured Mining

This type of mining is usually done to discover the structure of websites by collecting data related to the structure. Typically, it considers two types of connections: static and dynamic.

# **SERVER LOG**

## The server responds to user requests and the server log records all transactions from start to close of the server. They provide a time stamp for user requests and respond by registering the required ID with the required action.

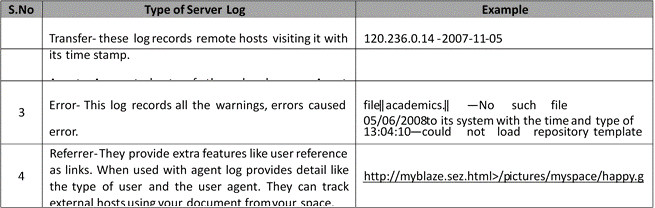
## These Log files can be placed in 3 places:

## • Web servers - An Internet server distributes web pages as required.

## • Proxy Server - A Proxy server is an intermediary calculator that acts as a computer hub through which user requests are processed.

## • Web Client - A Web client is a computer application, such as a Web browser, that runs on a user's local computer or workstation and connects to a server as needed.

## Table 1: Types of Server Logs



# **2.3.Mining the Web nowadays**

On most computer desks, the Internet Browser is the entry of information. The Internet browser is a key element in the study and use of access to information, using other applications and can be used more often and longer. Improving today’s access requires understanding how our enterprise takes over the Web, if someone else seeks to extract enterprise data on the Web to gain new insights.

Internet mining has been the subject of many studies, for example (Kimbrough, 2000), using e-commerce applications (B. Mobasher, Cooley, Robert and Jaideep Srivastava, 2000; B. Mobasher, Dai, Honghua, Luo, Tao, Miki Nakagawa, 2000; Padmanabhan, 2001; Sen, Padmanabhan, Tuzhilin, White, & Stein, 1998; Sen, 1998; Vranica, 2001), as electronics aim to make one tire even more.

However, online mining has been problematic due to poor and incomplete data analyzed. Typically, data is extracted from Web server registers, which have some drawbacks as a source of information. We do not describe the sources and for the site-centric data that records information on the Web server as visitors arrive at the site. Next, we will not discuss the techniques that combine web-centered service information with the help of third-party client handling if we are looking to use site user pages. We will then discuss methods to work with such as aggregation, nearest neighbor, and other envisioned techniques, that make to provide service information to better personalize. Following the start of user analysis and focus, at the center of the pages on the server, we will not end with the latest work analyzing the relocation and focusing on the use of customer service.

#### **2.4.Server-Side Techniques**

The widespread use of the Web among computer users makes Web analytics a potentially rich source of data for the study of user behavior and the creation of information. Because the Web Protocol (Hypertext Transfer Protocol, or HTTP) is designed for efficiency and scalability, it provides little information during a Web transaction (December & Ginsburg, 1996). This basic information includes the client's Web address, the requested page, the time of the request, and the size of the requested page, but not, for example, the user identification name. As information is transmitted from the client to the server, it is usually written to a log file to create a record of server entries.

Lack of user information is exacerbated by Web protocol statelessness: no memory is held between client site requests (December & Ginsburg, 1996). Therefore, without any additional programming, each link is re-established and consumer identity conclusions are made in a timely manner. Early papers have identified the usefulness and disadvantage of online server registers for merchants (Sen et al., 1998). The data readily available in web registers meet a significant portion of the information needs for segmentation of merchant behavior, while most information needs for demographic, geographic, psychographic, and profit sharing require information to be obtained from visitors.

Internet server logs present a distorted picture not only because some data never reaches the server, but also because the data being reported itself is inaccurate. Confused data arises because Web server logs record a client's Internet Protocol (IP) address. Due to the lack of unique IP addresses, most organizations use Dynamic Host Configuration Protocol (DHCP) to manage a set of IP addresses. Addresses are assigned to computers as needed, and so the same address can be assigned to two (or more) users in a row. Moreover, the same user can get different IP addresses on consecutive days, or even during one day.

Some of these shortcomings are addressed through the use of Proxy servers. Proxy servers act as connections between Web clients and servers, responding to client requests as if they were Web servers themselves. Proxy servers are installed by many organizations, including Internet Service Providers (ISPs), for a variety of reasons, including more efficient bandwidth utilization, faster response times, and security. In relation to Web mining, Proxy servers represent an improvement of Web server data because they consolidate Internet traffic from an entire group or organization. Thus, the Proxy server registry records all access to a group's site, regardless of the Web server destination. Because a typical Web user will visit multiple servers in a single session, the upstream data available from a Proxy server can lead to new insights into user behavior.

Although Proxy servers provide a richer database for those who use them, they also exacerbate the problems of those who work with Web server registries. Because the Proxy server switches itself between the client and the Web server, the Web server sees a single Proxy server instead of multiple clients. For example, the widespread use of Proxy servers by AOL does not allow Web servers to determine which AOL member is accessing a particular site. Many companies use Proxy servers for security reasons and thus all company employees are presented as the same Proxy server on a Web server.

Content protection is another problem in capturing accurate data for later mines. Because most websites change infrequently, Web browsers improve page access by storing a copy of the Web pages on the user's computer in a local cache. Later, if a user requests a saved page, the browser simply displays the saved version. (Using the Back button of a browser is a common case because the page we just viewed is usually in the browser's memory.) For extracting data on the Web, the problem is that the accesses of the saved pages are not logged by the server, since the browser simply submits its own copy, without accessing the server. In an extreme case, a user can take two pages from one server, then spend hours moving back and forth between the two pages. The Web server would record this after one page access was immediately followed by another, but successive switches between the two pages would not be logged.

#### Proxy servers add to the challenge because even Proxy pages are stored, so a browser may request an updated version of a page, and the Proxy will eavesdrop on the request and return its stored page - again preventing the Web server from register the request. In short, Web server logs suffer from several important drawbacks: network configuration issues Cloud server data, user browsing experiences are fragmented across servers, and lack of user context prevents analysts from examining user behavior before and after access.

#### **2.5.Cookies**

Due to problems in using network addresses to identify users, Web developers have sought tools to connect users to Internet access. The challenge is that the Web itself is stateless - meaning that there is no link between consecutive page entries by the same user. In 1995, Netscape Communications introduced the "cookie" as a mechanism to add status to Web client server sessions. (The original cookie specification was coded as RFC 2695 Internet Engineering Task Force (Kristol 2000).) With cookies, Web servers can store short strings of client information. For example, a database key or username is a common choice for cookie data. When the user returns to the same site, the previously written "cookie" information is automatically sent to the server with each access to the page. In operating systems that support multiple user profiles, cookies can reduce the hassle of using network addresses to represent the user by associating Internet traffic with a returning user, rather than just an IP number. Often, however, computers shared by many use a single login, enabling cookies. Other people turn off cookie functionality in their browsers in response to privacy concerns (Schwartz, 2001).

### CHAPTER 3

### Improving Workflow Management Support with Client-side Monitoring

In workflow systems, there is a tension between a highly structured workflow (with a robust, fundamental model) and the need to model review through synchronous and asynchronous group interaction. Building flexibility in workflow processes is the topic of many recent works, for example (Antunes, 1995) (Ginsburg, 2001) (Bernstein, 2000). Client monitoring allows us to go further in designing an effective and flexible workflow management system (WFMS).

Let us consider the general problem of a working group consisting of several participants. This working group is assigned a predetermined task (such as processing a document), which must be done in a structured way. Participants working on this task are called actors. These actors use typical productivity tools, such as other documents, websites, spreadsheets, etc, to complete the task. The typical order in which the task is done is for the first group of actors to perform a knowledge discovery, process the input with the knowledge gathered, and pass the output generated to the other actor in the course of the work.

There is a dark "black box" that connects successive versions - our colleagues work inside the black box. In the workflow literature, a Petri Net model (Weitz, 1998) is used which represents documents as signs following a path. The sign goes through different stages (or checkpoints), which are labeled V1 to V6. To advance the mark, knowledge discovery is done inside the black box. With the help of additional resources, actors (agents who move the document forward through successive versions) perform several tasks on target between two phases or checkpoints. Thus, we can say that each actor performs some knowledge discoveries by inputting a version (a sign) and then passing the generated output (again a sign) to the other actor (s) in the stream for additional processing. This chain continues until the task is completed.

We would like to enlarge the steps between successive versions. To unravel the mystery of the black box and shed light on the process of knowledge discovery, let’s analyze this black box in more detail. This black box contains some items like web pages, Word documents, Excel spreadsheets, PowerPoint slides and databases. These objects are used by actors as they perform knowledge discovery and process input (sign). If by any means we are able to trace the activities of the actors, viz. What objects they used, how much time they spent on each, what operations they performed on each - would be a big step towards unraveling the mystery of this black box. Awareness Actors ’canonical awareness (Brown and Duguid, 1991) can be exposed by client monitoring and is very useful in workflow management. Client monitoring, in fact, gives us information on how, when, why and which objects are used in knowledge discovery.

Therefore, we can document, analyze, and store them in the organization's knowledge bank, so that when a similar task is assigned to a working group in the future, this awareness would prevent for the group to recreate the steering wheel. Our Upcoming Developments section describes Custom User Interface techniques to allow synchronous interface changes when a member of a group discovers a new and useful information asset; we can derive usefulness from client actions (e.g. large sections copied and pasted from new source). Discounts can be a powerful addition to a WFMS to better inform group participants, who may be physically close or geographically dispersed, during the various stages of document preparation.

Traces of knowledge in the absence of coordination and awareness are simply ad-hoc, with no transfer from one session to the next. Thus, awareness helps actors perform informed disclosure of artifacts. Therefore, newly found artifacts must be registered to be used again. Customer monitoring is ideal for this purpose. Our proposal aims to create an infrastructure to capture this awareness. With Python client monitoring software, we can pipe client registers to a server for collection, pre-processing, and analysis to make pattern discovery in event logs.

***3.1.Web Usage Mining***

Web usage mining also known as Web registry mining is the application of data mining techniques to large Web registry repositories to discover useful knowledge about user behavior patterns and site usage statistics in Internet that can be used for various web design tasks. The primary data source for Web usage mines consists of text logs collected from multiple Web servers around the World. There are four stages to using a web application:

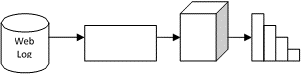
***Data collection:*** user registry data is collected from various sources such as server side, client side, Proxy servers etc.

***Pre-processing:*** Performs a series of Web registry file processing that covers data cleaning, user identification, session identification, path completion, and transaction identification.

***Model discovery:*** Application of various data mining techniques to process data such as statistical analysis, correlation, aggregation, model matching, etc.

***Pattern Analysis:*** Once patterns are detected from Web logs, uninteresting rules are filtered. The analysis is done using knowledge retrieval mechanism such as SQL or data cube to perform OLAP operations.

The four stages are described through Figure 3:



* Data Cleaning User Identification
* Session Identification Transaction Identification

on

* Model discovery
* Pattern Analysis

The purpose of this paper is to provide an overview of Internet usage utilization and a pre-processing phase study. The data collection section lists the different data sources, the pre-processing section reviews the various tasks done in identifying the session, the path completion process. The remaining sections summarize about model discovery, analysis, and various application areas where the Web usage mine is used.

**3.2.Data collection**

Data collection is the first step in the process of mining Web usage. It consists of collecting relevant Web data. The data source can be collected on servers, client sides, Proxy servers or retrieved from an organization database, which contains business data or consolidated Web data.

***Server-level aggregation*** collects client requests and are stored on the server as online registers. Online server logs are plain text that are independent of the server platform. Most web servers follow the usual registry format such as "IP address - username - password - date / time stamp - URL - version status code - sent bytes".

Some servers follow the Extended Registry format along with the referrer and user agent. ***The referrer*** is the URL reference link and the user agent is the string that describes the type and version of browser software used. Internet cache and IP address misinterpretation are two obstacles in the server registry. Online Cache keeps track of websites that search for and store a copy of those pages for a certain period of time. If there is a request for the same page, the cache page is in use instead of making a new request to the server. Therefore, these requests are not recorded in the log files.

***Cookies*** are unique IDs generated by the Web server for individual client browsers and automatically track the site visitors. When the user visits the next time, the request is sent back to the Web server along with the ID. However if the user wants privacy and security, they can disable the browser option for accepting cookies.

***Clear user login*** data is collected through registration forms and provides important personal and demographic information and preferences. However, this data is not reliably efficient for Web usage utilization.

***Representative level collection*** is the data collected by the intermediate server between browsers and Web servers. Proxy protection is used to reduce the loading time of a Web site experienced by users, as well as the network traffic load on both the server and client sides. The logs from the Proxy servers are of the same format as the Web server log and record the Web site request and response to the server. Representative traces can detect current HTTP requests from many clients on multiple Web servers. This can serve as a data source for characterizing the browsing behavior of a group of anonymous users who share a shared Proxy server.

**3.3.Data preprocessing**

The information available on the Web is heterogeneous and unstructured. Therefore, the pre-processing phase is a prerequisite for pattern detection. The purpose of preprocessing is to transform raw click-through data into a set of user profiles. Data preprocessing presents a number of unique challenges which led to a variety of heuristic algorithms and techniques for preprocessing tasks such as merging and cleaning, user and session identification, and more. Various research works are carried out in this preprocessing area for grouping sessions and transactions, which are used to discover user behavior patterns.

***3.4.Data Cleaning***

Data cleansing is a process of removing trivial items such as jpeg, gif files or sound files and references due to spider navigation. Improved data quality improves analysis on it. The HTTP protocol requires a separate connection for each request from the Web server. If a user requests to view a particular page, along with server registry entries, graphics and scripts are downloaded in addition to the HTML file. An exception is the art gallery page where images are most important. Check the status codes in the logs for successful codes. Status code less than 200 and greater than 299 were removed.

***3.5.User Identification***

Identifying and individually identifying their users make a website is a step in using the use of the Internet. Different methods must be followed for identification and users. The simplest method is to assign an ID to use different users at different IP addresses. In the Proxy service, many users are using addresses and once using multiple browsers. An Extended Recording format overcomes this problem by referring information, and a user agent. If the IP address is a user with the previous login and the user agent is different, then if the user is assumed to be a new user. If we do not have an IP address and the user agents are at the same time, then the URL refers to the creator and topology of the checked pages. If the requested page was not reached directly for another reason to be used in case the user identifies if it is used in our address. The caching problem can be corrected by using a short expiration time on HTML pages by running the browser to retrieve each page from the service.

***3.6.Session Identification***

A user session can be defined as a group of pages visited by the same user within the duration of a particular visit to a Web page. A user can have a single or multiple sessions over a period of time. Once a user is identified, each user's click stream is divided into logical groups. The method of splitting into sessions is called ***Sessionization*** or ***Session Reconstruction***. ***A transaction*** is defined as a subset of the user session that has a homogeneous page. There are three methods in session reconstruction. Two methods depend on time and one on navigation in Web topology.

***Time Oriented Heuristics :*** The simplest methods are time-oriented in which one method is based on the total time of the session and the other based on the time spent on a single page. The group of pages visited by a specific user at a given time is called the page browsing time. It varies from 25.5 minutes to 24 hours while 30 minutes is the time predetermined by R.Cooley. The second method depends on the time spent on the page which is calculated by the difference between the two time stamps. If more than 10 minutes pass, the second entry is assumed as a new session. Time-based methods are not reliable because users may get involved in some other activities after opening the website and factors such as busy communication line, time of uploading components to the website, size of content websites are not considered.

***Navigation-Oriented Heuristics :*** uses Web topology in graphic format. IT considers linking to the site, however it is not necessary to have hyperlink between two consecutive site requests. If a website is not linked to the site previously visited in one session, then it is considered as another session. Cooley proposed a referrer-based navigation-based heuristics in which the referrer URL of a page should exist in the same session. If no referrer is found, then it is the first page of a new session.

Both methods are used by many applications. To improve performance, different methods were invented based on time-oriented heuristics and navigation by different researchers. Various works have been done by researchers for the effective reconstruction of the sessions.

The referrer-based method and the time-oriented heuristic method are combined to perform user session identification within. The Internet access group is the set of all records in the Internet access register and are stored in chronological order. A set of user sessions is taken from the Web Access Log Set by following rules such as different users are distinguished by different IP addresses. If the IP addresses are the same, different browsers or operating systems indicate different users, and if the IP addresses are the same, different browsers and operating systems are the same, the referrer information is taken into account. The ***Referrer URL field*** is checked and a new user session is identified if the URL in the Referrer URL field has never been seen before, or there is a large interval (more than 10 seconds) between the entry time of this record and that of previous if the Referrer URL field is empty. If the sessions identified in the previous step contain more than one visit from the same user at different times, then time-oriented heuristics are used to split different visits into different user sections.

A simple algorithm was devised by Baoyao Zhou. An introductory session is created as a pair of URLs and the time required in a request sequence with a time stamp. The duration of a URL is estimated as the difference between the request time of the next entry and the current entry. There are no descendants for the last URL. So the duration is estimated as the average duration of the current session. The end time of the session is the time and duration of the start. This algorithm is convenient when there are more URLs in a session. The default time set by the author is 30 minutes per session.

***Smart Miner*** is a new method devised by Murat Ali and the team. This framework is part of Web Analytics software. Sessions built by SMART-SRA contain consecutive pages accessed by the server works in two stages and follow the Timestamp Ordering Rule and the Topology rule. In the first stage, the data stream is divided into shorter sequences of pages called candidate sessions using the rules of session length and page stay time. In the second phase, the candidate sessions are divided into maximum sub-sessions by the sequences generated in the first phase. In the second stage the limitations of the topology rule referent are added eliminating the need to insert browser movements backwards.

Pages without any referrals are defined in the candidate session by the Web topology. Then those pages are removed. If there is a hyperlink from the previously built session, those pages are added to the previous sessions. In this session are formed one after the other. An agent simulator was developed by the authors to simulate a current Web user. Randomly generates a typical Web site topology and a user agent to access the same by its client and acts as a real user. An important feature of the agent simulator is its ability to model the dynamic behaviors of a Web agent. The time limit is also considered as the difference between two consecutive pages is less than 10 minutes.

Another method using ***Integer Programming*** was proposed by Robert F.Dell. The advantage of this method is the construction of all sessions simultaneously. He suggests that every Web registry be considered a ***registry***. Logs from the same IP and agent address as well as affiliates are grouped to form a session. A binary variable is used and a value of 1 or 0 is assigned depending on whether the registry has been assigned a position in a particular session or not. Restrictions such as each register are used at most only once and only in one session for each listed position. A maximization problem is formulated.

To improve the solution time, the binary variables subset is set to zero. An experiment was performed to show how the objective function changes and the results are obtained with raw registers and filtered for MM objects and errors. Unique pages and links between pages are numbered.

Graphs are also used to identify the session. It gives more accurate results for identifying the session. Websites are presented as vertices and hyperlinks are presented as edges in a graph. User navigations are modeled as crossings from which frequent patterns can be detected. i.e., sub-permeations contained in a large permeability ratio. A method was proposed by Mehdi Heydari and the team. They considered that client side data is also important to reconstruct the user session.

There are three stages to this method. In the first phase an AJAX interface was created to monitor user browsing behavior. Events such as Start Session, End, Page Request, Page Loading, Page Focus are created along with user interaction and logged in session. In the second phase a basic graph is constructed using data for Web usage. Web browsing time is indicated as a peak. Skip is a sequence of consecutive pages in a base graph. A database with traverses is created. In stage three, the graph utilization method is applied to the database to detect the frequently weighted model. The most commonly weighted model is the model when the throughput weight is greater than or equal to a ***Minimum Browsing Time***.

Another algorithm has been proposed by Junjie Chen and Wei Liu in which the session data and identification are cleared. In this delete foreign content for mining algorithms collected from Web logs. The user activity log is checked, it judges whether the record is a spider record or not and it judges whether it is embedded in the page or not according to the URL of the requested pages and the graph of the page structure. Session registration is required if no session exists, a new session is set. If the current session ends or exceeds the preset time threshold, the model will end it and find a new one. Graph utilization methods build accurate sessions and the time taken is also relatively less. More research needs to be done in this area.

***3.7.Path Completion***

Sites may be missing after building transactions due to Proxy servers and caching problems. So the missing pages are added as follows:

The site request is checked whether it is directly related to the last page or not. If there is no link to the last page, check the recent history. If the registry entry is available in recent history, then it is clear that the "back" button is used for memorization until the page is reached. If the referrer register is not clear, the site topology can be used for the same effect. If multiple pages are linked to the requested page, the nearest page is the source of the new request and so that page is added to the session. There are three approaches in this regard.

***Reference Length Approach:*** This approach is based on the assumption that the amount of time a user spends on a page is related to whether the page is a help page or content page for that user. It is expected that the time spent on the help page will be small and the content page will be more. A reference length can be calculated that estimates the disconnect between auxiliary references and content. The length of each reference is estimated by taking the difference between the time of the other reference and the actual reference. But the last reference has no other reference. So this approach assumes that the latter is always an auxiliary reference.

***Maximum Forward Reference:*** A transaction is considered as the set of pages from the visited page until there is a backward reference. Forward referral pages are considered as content pages and the path is taken as index pages. A new transaction is considered when a backlog reference is made.

***Time Window:*** A time window transaction is compiled from triple IP address, user identification, and the duration of each Web page up to a limit called the ***time window***. If the time window is large, each transaction will contain all page references for each user. The time window method is also used as a merge approach in conjunction with one of the previous methods.

An optimal algorithm was devised by G.Arumugam and S.Suguna to generate accurate path sequences using the entry history list based on the duplicated structure to design a complete path with optimal time. Server pages are required in this tree structure. There are two problems in this search as backlog referencing consumes more time on unused pages as well and pages which are referenced directly by the other server leading to incorrect session identification. To overcome these issues the authors give different algorithm. We will examine the data structures of the authentication algorithm such as Array List to represent Web Logs and User Access List, a Hash table to represent server pages, also use a detached structure in two directions. The two-way detached structure is used to store the Access History List (AHL) to represent the sequence of the user-accessed page.

Two hash tables are used, the main and secondary hash tables, in which the primary hash is used to store the sessions and the secondary table indicators that have a full path navigation. To resolve time consumption, only visited pages are saved in the login history list and are not considered unused. Using a single search in the history list, the page sequences are located directly. When pages are referenced by other servers they start directly from the page and not from the root. If the page is not available in the current sessions, start a new session and we can conclude that this is not a backlog, but the page has been browsed to another server. This method generates more accurate paths than the maximum forward methods and reference lengths.

**CHAPTER 4**

**DATA ANALYSIS PART**

**Analysis of Web server log by Web usage mining for extracting users patterns**

The WWW is a system of hypertext-related documents accessed through the Internet. About 11 hundred million people access the Internet every day. And so the information available on the WWW is also growing. With this ever-increasing information and proliferation of Internet services and Web-based information systems, websites are also growing to host them. Before analyzing such data using the data mining technique, the server registry must be processed in advance.

Log file data provides insight into the use of the website. They can be collected from Web servers, representative servers, web client. Web Mining applies the technique of extracting data to extract knowledge from these Web registry files. This paper discusses Log files and uses online mining techniques to derive usage patterns using WEKA.

**Keywords**: Pre-Processing, Web Usage Mining, Web Server Log Data, Classification, Clustering, Rule Based Mining, Pattern Discovery

**4.1.Pattern discovery and analysis**

Once user transactions have been identified, a variety of data mining techniques are performed to detect the pattern in the mines using the Web. These methods represent the approaches that often appear in the data mining literature such as the detection of association rules and subsequent patterns and grouping and classification and so on. Classification is a supervised learning process because learning is driven by the assignment of instances to the classes in the given training. A data item is mapped to one of several predefined classes. It can be done using inductive learning algorithms such as decision tree classifiers, naive Bayesian classifiers, Support Vector Machines, etc.

The ***Association Rule Disclosure Techniques*** are applied in transaction databases where each transaction consists of a set of items. Using the A priori algorithm, the largest groups of frequently accessed items from transaction databases are detected. Clustering is a technique for grouping users who display similar browsing patterns. Such knowledge is especially useful for highlighting user demographics in order to perform market segmentation in E-commerce applications or provide personalized Web site content. Sequential patterns are used to find cross-sectional patterns, such that the presence of a set of items followed by another item in a set of timed sessions. Using this approach, online marketers can predict future visit patterns which will be useful in placing ads targeting certain groups of users.

***Model analysis*** is the final stage of utilizing Web usage. Mined models are not suitable for interpretations and judgments. So it is important to filter out uninteresting rules or patterns from the group found in the pattern discovery phase. At this stage tools are provided to facilitate the transformation of information into knowledge. The exact analysis methodology is usually guided by the application for which the mine is made. Knowledge query mechanism such as SQL is the most common method of model analysis. Another method is to load usage data into a data cube in order to perform OLAP operations.

**CHAPTER 5**

**IMPLEMENTATION PART**

**Contents of Log File**

A server registry file is a registry file that automatically creates and maintains activities performed on it. Keeps a history of page requests. Helps us understand how and when our Web pages and application are being accessed by Web browser. These registry files contain information such as the IP address of the remote host, the requested content, and the time of the request.

## SYNTAX

IPaddress, logproprietor, Username, [DD:MM:YYYY: Timestamp GMToffset] , "req method" .

### *Ex*

104.11.13.108 - - [13/Jul/2021:16:56:12 -0600]

"GET /EDC/cell.htm HTTP/1.0" 200 4093.

* **IP address**- The IP address of the http request is recorded to identify the remote host. Ex: 204.31.113.138.
* **log proprietor**- The name of the owner making an HTTP request is recorded through this field. They do not expose this information for security purpose. When they are not exposed they are denoted by (-).
* **Username**- This field records the name of the user when it gets a http request. They do not expose this information for security purpose. When they are not exposed they are denoted by (-).
* **date***-* Request date is recorded here in the mentioned format. Ex: 13/Jul/2021.
* **time**- Request Time of the HTTP request is recorded here in astronomical format. Ex: 16:56:12.
* **GMToffset**- This field shows the time difference between the actual request time and Greenwich Mean time so that request from corner of the World can be analysed in any part of the World. Ex: -0600.

**Reqmethod-** The request type of the request is stored. Ex: GET. Types of Log Formats

* NCSA Log Formats;
* W3C Extended Log Format;
* Microsoft IIS Log Format;
* Sun One Web Server Format.

## 5.1.NCSA Log Formats

National Centre for Supercomputing Application (NCSA) established in 1986 developed a Web server called HTTPD at its centre. This Web server had a log initially which had several extensions later.

## NCSA Common Log or Access Log Format

Stores basic information about the request received.

## Syntax

Host IP address, Proprietor, Username, date: time, request method, status code, byte size.

### *Ex*

200.40.12.4, -, -, [2021/Jul/10:10:16:52 +0500], GET /svec.html http 1.0‖, 200, 1460.

## NCSA Combined Log Format

Stores all common log information with two additional fields. ―referrer‖ –history of request, ―user\_agent‖ – type of browser.

## Syntax

Host IP address, Proprietor, Username, date: time, request method, status code, byte size, referrer, User\_agent,

Cookie.

### *Ex*

200.40.12.4, -, -,[2021/jul/10:10:16:52 +0500], GET /svec.html http 1.0‖, 200, 1460, [http://www.cbci.com/,](http://www.cbci.com/)

―Mozilla/5.0 (WIN:7)‖, ―UserID=om123;Pwd=101112‖.

## 5.2.NCSA Separate Log Format

In this type the information is split into 3 log files instead of storing it in a single file.

The three log files are:

* Access log
* Referral log
* Agent log

## 5.3.W3C Extended Log Format

The Worldwide Web consortium (w3c) is an international standards organization. They provide rich information hence the name extended log format.

The lines starting with # contain directives. #version <int><int>

#Software- the software which generated the log . #date-<date><time>

#fields-this directive lists a sequence of entries.

They are as follows:

## Table 2: W3C Directives

**Acronym Description**

C Client

S Server

CS Client to Server

SC Server to Client

r Remote host

Sr Server to Remote host

rS Remote host to Server

**Ex**

#version: 2.0

#Software: Microsoft windows server1.0 #date : < date> <time>

#field: C-ip S-ip CS-username CS-method CS-uri CS-version CS-user-agent SC-status

2021-04-14 10: 16: 42 , 192.16.14.1, 200.14.100.4 - GET/Mypictures.gif http1.0 ―Mozilla/4.0‖ 200.

## 5.4.Microsoft IIS Log File

Internet information service enables us to track or record the activities that take place on our website through File Transfer Protocol (FTP), Network News Transfer Protocol (NNTP), Simple Mail Transfer Protocol (SMTP) allowing us to choose a log format which works in sync with our system environment. Some additional attributes given are as follows: Time elapsed, total bytes transferred, target file.

## Syntax

IP add, date timestamp, Server name, Server IP, elapsed time, http request size, byte size, status code, error, request method.

### *Ex*

192.16.10.1,-,10/4/01 14:02:10, svec, 170.42.14.2, 1604, 140, 4240, 200, 0, GET, /Mypicture.gif.

## 5.5.Sun One Web Server

They are similar in functionality with the above mentioned log formats. But provides more security in 2 ways.

-By using Secure Socket Layer (SSL) between client and server.

-Administrator can provide access controls or permissions to files and directories.

## 5.6.Request & Response by the Server

All the log formats specified above has fields that record HTTP request in the form of elapse time and response in the form of status code.

# **To Handle Request**

## Authorization

Checks user ID and password

## URI Translation

Translates the Uniform Resource Identifier to local system path.

## Checking

Checks the correctness of file path with user privileges. MIME type checking:

Checks the Multi-Purpose Internet mail Encoding of the requested resource.

## Input

Prepares the system for reading input.

## Output

Prepares the output for client.

## Service

Generates the response to client

## Log Entry

Record the activity into the Log.

## Error

This field is used only if any of the above mentioned field fails from its normal execution. They are of 2 types.

They are as follows:

## 5.7.Connection Errors

They happen when a connection established for communication with the Web server drops. They are classified as follows:

## Void URL

This simply means that the format of the Uniform Resource Locater is invalid.

## Host Not Found

This error occurs when the Server could not be found with its host/domain name.

## Time Out

When a connection could not be established with in a predetermined time this error occurs. The default time out is set to 90 seconds.

## Connection Refused

This error occurs when an identified host refuses connection through its default port.

## No response from Web Server

When an identified Web server fails to respond with a time period this error has said to be occurred.

## Unexpected Error

These are errors that does not report itself in an anticipated manner; it cannot be classified into one of the predetermined categories.

# **To Handle Response**

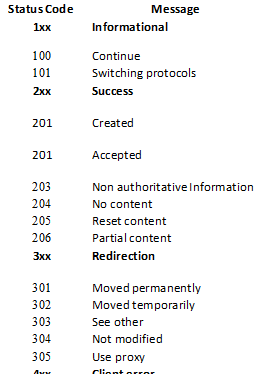
If a connection is established successfully with a Web Server then the Server responds with one of the following status codes:

## Web Server Status Code & Messages

The Status-Code element is a 3-digit integer result code of the attempt to understand and satisfy the request.

Table 3-clearly explains the message status.

**Table 3: Status Codes**



**Table 3:Contd…**

406

407

408

409

410

411

412

413

414

415

**5xx**

501

502

503

504

505

Not acceptable

Proxy Authentication required Request timeout

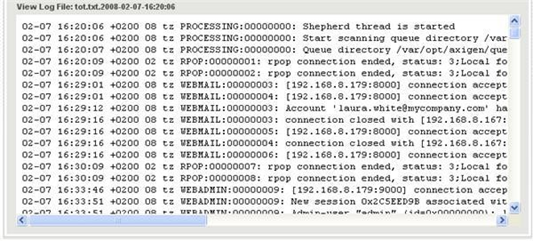
Conflict Gone

Length Required Precondition Failed Required entity too long Required URI too long Unsupported media type **Server error**

Not implemented Bad gateway Service unavailable Gateway timeout http version timeout

# **5.8. Experimental results**

A company’s log file in Figure 4 is analyzed using WEKA. We apply classification and clustering technique in it.



## Figure 4: Log File

(Username, ReqType and UserAgent of the Log File are Not Considered here for Mining)

By applying **If Then classification** [10] rules we obtain

## Rule

IF

freemem <= 193.5

freemem > 117.5

## THEN

**usr = 93**

+ 0.0001 \* outtime

- 0.0016 \* intime

- 0.0019 \* bytesize

+ 0.0022 \* req

- 1.9716 \* exec

## Rule: 2

IF

freemem > 304.5

fork <= 1.095

bytesize > 1626.5

## THEN

**usr = 88**

+ 0.152 \* intime

- 0.0009 \* outime

- 3.0119 \*req

- 0.0011 \* exec

+ 0.31 \* freeswap

Similarly around 17 rules can be mined. Applying ***kmeans*** we obtained prior probabilities of clusters.

Mean Distribution

Attribute: UserId

Normal Distribution. Mean = 25.1373 StdDev = 60.3306 Attribute:In Time

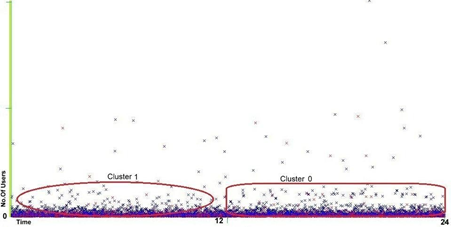
Normal Distribution. Mean = 16.3085 StdDev = 33.4104 Attribute: Out Time

Normal Distribution. Mean = 2979.7651 StdDev = 1538.001 Attribute: Byte Size

Normal Distribution. Mean = 271.0738 StdDev = 215.9147 Attribute: Request

Normal Distribution. Mean = 2.3778 StdDev = 2.7902 Attribute: Exec

Normal Distribution. Mean = 3.7507 StdDev = 6.2205



## Figure 5: Kmeans Cluster-Distribution

Cluster: 0 Prior probability: 0.6381

Cluster: 1 Prior probability: 0.3619

From the above graph Cluster 0 contains maximum number of users.

## Farthest First

Farthest first is a variant of K Means that places each cluster centre in turn at the point furthermost from the existing cluster centre. This point must lie within the data area. This greatly speeds up the clustering in most of the cases since less reassignment and adjustment is needed.

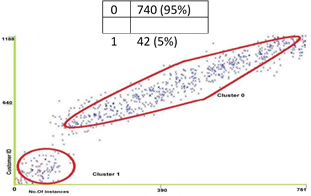
## Cluster Centroids

Cluster 0

1098.0 Antoneta M Other password 10/7/2021 10/7/2021 2 Cluster 1

123.0 Ana F Mobile 9.940870123E9 9/1/2021 9/1/2021 227 Time taken to build model (full training data) : 0.06 seconds

Clustered Instances



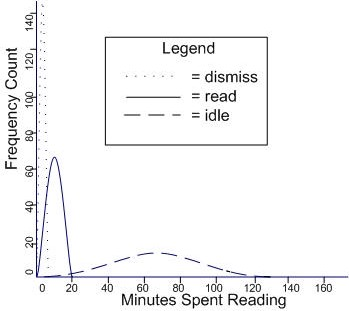
**Figure 6: Centroids of Cluster**

**CHAPTER 6**

#### **Server-Side Programming to Augment Basic Server-Side Data**

Although cookies improve the quality of data by the server, the incorporation of cookies still maintains a site-centric perspective. The data still needs to be analyzed site-by-site, which means that cross-site behavior cannot be observed. Moreover, user actions on the client are visible only through the Web browser preventing analysts from observing patterns between several applications. Web browsers can be added with either server-side software or embedded scripts to provide additional information on Web servers.

Lamacchia (LaMacchia, 1996) modified the free Excite search engine to build an active search agent in its "Internet Fish" system. Ginsburg (Ginsburg, 1998) and Ginsburg and Kambil (Ginsburg & Kambil, 1999) modified the same search engine to provide document reader time within a full-text search framework in the "Annotate" system. In the work, a Gaussian trimodal reader time distribution was found as shown in Figure 7. The reader time distributions shown in Figure 7 lead us to the conclusion of three types of document readers:



**Figure 7 – Counts of reading time (minutes)**

Very short browse time (µ = 3 minutes) – Document uninteresting, and is immediately dismissed. The next document is chosen from the search results or the session is exited

Medium browse time (µ = 10 minutes)

– Document interesting enough to read for more than a few, but less than about 20, minutes.

Long browse time (µ = 65 minutes) – User is idle, having either left her computer or is doing other tasks at the computer. The mean time for this Gaussian distribution was about 65 minutes.

The conclusions above are inferences, and while reasonable, they cannot be verified without contextual information.

With client-side monitoring that stretches across all the commonly used applications, however, analysts could resolve the ambiguity in long browse times. In a cross-application monitoring environment, if they are still active on their PC or if they have gone completely idle.

***6.1.“User-Centric” Surf Mining***

In addition to technical barriers that limit the accuracy and richness of server logs, server-side data suffers from a lack of data in the user context. As (Padmanabhan, Zheng, & Kimbrough, 2001) point out, Web server data is essentially incomplete. Although many users simply browse the Web to spend time, others access the Web for specific purposes.

For example, a user who sends an email about an upcoming restaurant reservation may want to attach a map to the email, and so will use its browser to search for the restaurant address and then submit it to a map page to generate maps. With server-side log analysis, the link between e-mail, phone search, and map search is lost. By extending analytics by the client to monitor applications outside the browser, data mining can reveal patterns of user behavior around Internet access, rather than just within it. This field has been very active in both academia and industry.

Researchers are struggling with incomplete and large amounts of data; a major focus is pre-processing to make better analysis and integration (B. Mobasher, Dai, Honghua, Luo, Tao, Miki Nakagawa, 2000). Another focus is user labeling the session. For example, in Kimbrough et al., (Kimbrough, 2000) the authors provide a framework of usage measurements and analyze the surfing behavior of 30,000 users in an online travel booking area. To do the study, they used user-centric click-through data provided by a leading market data vendor.

In Damiani et al. (Damiani, Oliboni, Quintarelli, & Tanca, 2001) the authors present a new user modeling technique based on a temporary graph-based model for semi-structured information in order to formalize the user trajectory, but no techniques are suggested. software specific. Edelstein (Edelstein, 2001) talks about the challenges of click analysis (consolidating raw data from Web server logs across multiple servers and turning them into usable registers).

Another problem is that logic servers can be multiple physical servers for load balancing or geographic reasons. The logs are difficult to clean in some cases. It is not always easy to remove foreign items, or to identify users and sessions, or to identify transactions. All of these make it difficult to integrate this data with other data sources.

#### **6.2.Generic Agents to help Knowledge Work**

The work presented by (Schnurr, 1999) is not specifically addressed for any software architecture. The authors propose an abstract agent to help integrate the semantics of semi-structured documents (e.g. XML) and business processes in order to support end-user "desk work". The goal is to provide a proactive reasoning agent and improve knowledge management. A similar discussion has been presented by Abecker, et al. (Abecker, Bernardi, & Sintek, 1999). The authors discuss the need to actively support the user in a search task. They propose the principle that the organizational memory system should actively provide interesting insights.

They also argue that applications that combine data, formal knowledge, and informal representations all have embedded knowledge.

Thus there is a need for a ‘common view and use of all these representations’ (Abecker et al., 1999). Their most ambitious idea is that normal work should not be disturbed (the principle of distrust) --- the system should observe the user doing tasks, and then 'automatically collect and store interesting facts and information in a modest way' (Abecker et al., 1999). Their application is called ***KnowMore***, and contains workflow support, and support for heterogeneous IR data through homogeneous knowledge descriptions.

#### **6.3.Improving Personalization and Recommender Systems**

One of the problems faced by recommendation system personalizers and designers is incomplete and inaccurate data regarding customer preferences and transaction histories. As discussed (Padmanabhan et al., 2001), personalization by Web server registry analysis alone is likely to be erroneous. A richer data source is needed to personalize effectively or provide strong recommendations. Adomavicius and Tuzhilin (Adomavicius & Tuzhilin, 2001) propose a multi-dimensional “hypercube” technique and OLAP to provide a richer recommendation framework (they do not define the exact functional form). However, they do not provide any software architecture related to data collection (how to fill the hypercube). We will discuss the start of a new generation in data collection, customer monitoring system.

#### **6.4.Client-Side Monitoring Today**

The “next generation” of monitoring recognizes the inherent limitations of server-side conclusions, data mining, data filling, and data manipulation. Instead, recent research has turned to adding software to the client to give a more detailed view of session activities. Such client data can be usefully combined with server data or third-party auxiliary data for a more complete user profile.

The tip of the iceberg is in a paper on personalized purchases (Jörding & Michel, 2001); the paper presents a system, TELLIM that generates personalized multimedia presentations during execution time. It matches the client display. To do this, the system monitors client interactions; for example starting an audio or video player. The system then concludes if the end user is interested in different elements of the presentation. Once a learning algorithm is implemented, only the preferred presentation elements are provided moving forward, and the repressed items are shown only as links. The technology uses the combination of dynamic HTML (DHTML) and a Java applet to monitor events on the client.

# **6.5.Goals for Client-Side Monitoring**

# Given the strengths and weaknesses of server-side monitoring, we have outlined four goals for a client-side monitoring framework: monitoring should not interfere with user activities, data collection should be user-specific individual, developers should be able to easily specify which data is collected and how, and finally, the framework should be extended to include monitoring of other applications by the client.

# Non-interference in user activities is already true for server-side recordings, but shifting to the client side makes distrust a key element. If users are asked to take additional actions to support monitoring, they will look for ways to reduce the burden of doing so, and are likely to distort monitoring results. Thus, the goal becomes to monitor background actions as users work with Web client applications. However, distrust must be balanced against the need to collect data. To an extreme, customer monitoring may require users to comment on every action they take, including the links they review, but choose not to follow, going back to previous pages and all other actions. Although such monitoring would generate rich data (at least for a short period), it is clearly unstable.

# One of the main weaknesses of server-side monitoring is that user identities are usually lost on the Web server. (If a Proxy server is employed, then the Proxy server will usually have access to the user data, but the Proxy will certainly disguise the user identity when logging in to the requested Web server.). By recording user behavior on the client , it is possible to link actions to users much more closely. For example, on the client side, login and user profile information is available in addition to the network address information sent to the client. Using Proxy servers, a significant problem for the Web server registry mine, is not a problem with client monitoring.

# Closely merging user identity and user actions, although it helps in data mining, raises issues of user privacy. With so much data available to the client, recording and mining data can reveal extensive personal information. In an organizational environment, this may be appropriate, as companies may want to know how employees are using company resources. In other settings, however, such extensive data collection can be problematic. In such cases, cryptographic techniques can be applied to maintain data integrity while disrupting the connection between the data and a known person.

# The breadth of data available to the client poses another problem for those who design solutions on the client side: what data needs to be collected ? At the client, every mouse movement, every keystroke, every click and more are available for recording and analysis. The problem of what to collect is not just a matter of granulation, but also of concentration. For example, within a session on the Web, should the system login link stay where a user hovers over a hyperlink but does not click ? Or should only the hyperlinks followed be noted ? Which actions need to be recorded depends on what the analyst wants to draw, of course. The range of candidates has an important implication for monitoring the design of the framework: the framework should allow framework users to easily specify what data is needed.

# Finally, users enter as part of a broader context. Thus, the ideal monitoring framework should be scalable to include additional applications; ideally, office productivity applications would be included. A complete client-side monitoring solution can record user actions continuously in many applications. Integrated recording can support new conclusions about user behavior. For example, a user writing a memo may search the Internet for background information by copying useful information into the word processor that he / she is using to write the memo. Without knowing what information is copied and pasted, there is no way to distinguish between all the pages the user has browsed. Once the word processor is included in the framework, however, data extraction by the client can reveal which Web resources are most often copied and pasted into documents. The implication is that those resources that are copied most often provide the most relevant information to a particular user.

# In summary, an ideal monitoring framework would address several goals: non-interference, close association between recorded data and users, flexible data collection, and the ability to integrate monitoring with other client applications.

# **6.6.Framework for User Monitoring**

Ideally, a common Web monitoring framework could be developed for use on many different platforms. Unfortunately, the tight integration required between a monitoring framework and user applications precludes a single framework. The widespread adoption of Microsoft Windows, however, means that a Windows solution addresses the needs of many framework users. Although the previous client monitoring discussion applies to all platforms, the implementation we present is based on Microsoft Windows. Additionally, the proposed framework is currently built around Internet Explorer, but can be extended to other browsers. Current applications of the proposed framework integrate Microsoft Model Component Object Model (COM) technology (Box 1998; Microsoft Corporation 1995), Internet Explorer (IE), and Python (an open source programming language) (Python Language Website 2001). In short, the framework uses COM to capture events released by Internet Explorer and to record these events on the client for later transmission to a server.

In addition to significant market share, Microsoft Windows also provides the technological infrastructure needed to support a monitoring framework in its Component Object Model (COM). Most Windows applications expose some, if not all, of their functionality through COM interfaces. In other words, most of what users can do through an application's user interface programs can do through COM. In Windows, COM provides the infrastructure needed to allow applications to control applications (automation, in COM terminology) and to receive event notifications from applications. One of the main reasons for using COM is that automation and event notification occur in the background; users can continue to use applications as they normally do, while developers can use COM to control applications through a second channel.

Since the COM standard does not specify which properties or events to exhibit, there is no consistency between the monitored applications. Microsoft offers similar support across Office suite applications within one announcement, but because each software vendor is free to expose what they choose, researchers interested in monitoring the flow of information across multiple applications often have to be creative.

As shown in **Figure 8**, Office applications also support different events and features.

- Microsoft Word uses documents while Microsoft Excel uses worksheets and workbooks, for example. In our work, we are interested in tracking information flows within and between applications. Often users will search the Web for information through a Web browser, and then copy text (or simply the URL of a relevant Web site. Monitoring the pages they visit in Internet Explorer, as well as the use of Clipboard users Windows, we can study how people create documents and what resources are consulted in creating these documents.Of course, the same monitoring techniques can be used to gain a different perspective just for using the Web by recording information that does not are usually available through server-side data collection.

## 6.7.Monitoring Internet Explorer

As a Microsoft Web browser, Internet Explorer (IE) provides excellent support for controlling and handling events through COM. Thus, if users of interest use IE as their browser, the framework can gather extensive information from the browsing sections.

Although the NavigateComplete2 event lights up whenever the browser enters a Web server, it also lights up whenever a Proxy server eavesdrops on the request, each time the Back or Forward keys are pressed, and whenever users click the refresh button. Thus, NavigateComplete2 provides a much more complete picture of user browsing behavior than is possible with data collection by the server.



Microsoft Internet Explorer 6

*Monitored user actions:* NavigateComplete2, StatusTextChange, Quit *Monitored properties:* Document title, Visited URLs, Status text

Microsoft Word XP

*Monitored user actions:* DocumentChange, WindowSelectionChange, Quit *Monitored properties:*

Document path, Document name

Microsoft Excel XP

*Monitored user actions:* NewWorkbook, SheetChange, SheetSelectionChange, WorkbookBeforeClose, WorkbookOpen

*Monitored properties:*

Document path, Document name

*Other,*

*unmonitored applications*

Windows Clipboard *Monitored user actions:* Copy, Cut

*Monitored properties:* Clipboard contents (Multiple formats)

Our monitored version of Internet Explorer also tracks changes to the status text displayed by IE. (In Internet Explorer, the status text appears in the lower-left corner of the window frame.) Although scripts from the client can change the status text, IE itself changes the status text when the mouse pointer hovers over a link on a Web page. - the status text has been updated.

***Figure 8*** — Some actions and properties exposed through Microsoft's Component Object Model.

Additionally, the status text changes again as the mouse pointer moves from the link. By comparing the time stamps of successive status text changes, analysts can draw conclusions about the duration of link hover, perhaps by giving an idea of ​​which links users consider as follows.

Although the monitored version of Internet Explorer that we implemented captures very useful information, it does not currently record an important component of Web behavior: filling out Web forms. The NavigateComplete2 event is not useful for capturing form data because it only lights up after the page is loaded - we can capture the page URL containing the form and the page resulting from the form handling scenario, but not the form data itself .

Internet Explorer provides analysts with access to form data, however, through the BeforeNavigate2 event. As its name implies, this event lights up just before a new page is requested, and event information includes (among other things) form data (if we use the POST method - when we use the GET method, the data is encoded in the URL).

In addition to events, Internet Explorer also exposes properties through COM. The defined features are in addition to any information that is available through the events (such as the form data mentioned above). Although most of the properties are details about the size and placement of the window, some of the properties provide important information about the content. For example, the Document property (Microsoft Corporation 2001) corresponds to the representation of the Document Object Model (DOM) of the World Wide Web Consortium (World Wide Web Consortium 2000) of the current document. The W3C DOM provides extensive information useful for analysis, including document title, last modified date, links, and HTML page structure.

In addition to the information reported by Internet Explorer itself, the monitoring framework has access to data related to the current Windows session: computer network (IP) address, current time and time zone, current username, and more. Thus, any uploaded Web page can be logged and sealed with current time to support post hoc analysis of event logs.

### 6.8.Link mining

Because Internet Explorer (IE) exposes the document object (DOM) model of the current Web page, full page content is available. The IE DOM implementation includes some visible collections as well, including all links to a page, all IMG (image) objects, and all named anchors, among others. With the full DOM, extracting log files can compare link hovers (perhaps suggesting link scrutiny) with tracked links to identify the distinguishing features of tracked links. Path analysis may reveal that a user considered two links, followed one, then used the Back Button to return to the original page and followed the second link, then proceeded further, suggesting that the user was "tricked" by the descriptions. links to the original page in selecting the least relevant link.

Once post-processing has identified which links the user has considered, an analyst can compare and contrast three groupings: links that are not considered, links considered but not followed, and links followed. Identifying patterns between the three groups can help webmasters redesign pages to improve accessibility (or increase sales on an E-commerce site). A second order analysis may include the reference page or page suggesting so different pages can be generated dynamically based on the reference page.

Link mining can also be the basis for new visualizations. Currently, many Web graphs present it as static, relying on fixed links embedded in pages to demonstrate the relationship of links between pages. With the above link analysis, analysts can view the Web not as it is built, but rather as used. Visualizations can describe trajectories through the Web and help analysts identify patterns such as links followed only to get users back to the link page to follow another link under consideration. In addition to extending the potential benefits of online mining, the framework described here could be an integral part of other applications as well.

## 6.9.Monitoring Microsoft Word

Since documents are a common endpoint of information gathering tasks, monitoring a user's word processing application is important to understanding the flow of information as well as the use of the Internet. For example, although Internet browser monitoring will reveal which pages a user has visited (as well as considered links and other information as described in Internet Explorer monitoring), it does not tell page creators how they are information is used. Website designers (or web researchers) may want to know which text from a site's website is being included in the end-user documents, or which URLs are pasted into the documents.

The current version of the framework captures the listed information: events to discover when a user opens or creates a document or simply switches between several editable documents (DocumentChange), when the user leaves Word (Quit) or perhaps most importantly, when text is selected in active document changes (WindowSelectionChange). The last event is critical because a user pasting text into a document triggers the event. Because users are more likely to copy large blocks of text using the clipboard than to type again, knowing when a user pasted text (and recorded pasted text) is important to support Web usage analysis. Since Word also provides the name and path of the document, recording the aforementioned events can be accompanied by separate documents over time.

### 6.10.Compositional analysis

Another advantage of our monitoring framework is the ability to share a new version of the document to determine which parts are taken directly from the sources of information consulted, for example websites. We may also discover lesser indications of interest, for example a link to a Web site copied to the footnote of a Word document but none of its relevant contents.

The composition of the document, previously impracticable, has far-reaching implications for the knowledge management of the enterprise. Now we can quickly learn, among such common document genres as daily economic reports, strategy visions or ad-hoc memoranda, what sources of information they nurture in the creation process. Analyzes can be performed on a genre-by-genre basis, as well as on a job-by-job basis to better understand the workflow of the enterprise document. The analysis will be valuable to enterprise librarians as a metric to assess the usefulness of subscriptions, such as the Gartner and Seybold reports.

To perform document composition analysis, we store the material copied to the Clipboard in an important, value structure. The key in this case is to digest the hash of the copied text, and the value is the test itself. When the material from the Clipboard is pasted into a document, we store the pasted text in a similar data structure. So when the next version of the document is ready, we just need to convert the document to Rich Text Format or ASCII (a format suitable for string scanning) and then compare the text of the document with the material attached to the steps leading to the new version.

Traces of knowledge in the absence of coordination and awareness are simply ad-hoc, with no transfer from one session to the next. Thus, awareness helps actors perform informed disclosure of artifacts. Therefore, the newly found artifacts must be recorded for re-use as suggested long ago in Vannevar Bush's historical article on "Memex," his hypothetical hypermedia computer (Bush, 1945). Customer monitoring is ideal for this purpose. Our proposal aims to create an infrastructure to capture this awareness. With Python client monitoring software, we can retrieve client registers on a server for collection, pre-processing, and analysis to make pattern discovery in event logs.

## 6.11.Monitoring Microsoft Excel

Although Excel spreadsheet capabilities do not play a significant role in our current search, we have instrumented Excel to detect user actions similar to those captured in Microsoft Word. For Microsoft Excel, our current application records when users create or open workbooks (Excel files that many Excel users are familiar with) and when they create or switch between worksheets Excel books). Like Word, Excel supports notification of a selection change (SheetSelectionChange), which notifies the framework when an Excel user has pasted an item into a worksheet.

## 6.12.Windows clipboard: The link among applications

One of the critical challenges in implementing a framework for cross-application monitoring is not only capturing user actions within common applications, but across applications as well. Microsoft Windows offers many different mechanisms for applications to communicate with each other, including the Component Object Model (COM), Object Linking and Embedding (OLE), and many more. However, many Windows users come back to copy and paste to transfer text and graphics from one application to another.

As discussed above, both Word and Excel support event notification when a user inserts an item from Windows memory. However, such notice does not report the attached data itself or the data source (the application in which the cutting or copying was performed). Moreover, there are thousands of Windows applications, far more than we can hope to instrumentalize, even given the ease of implementation of the current framework. However, like cross-application interoperability, integrating the Windows clipboard into the monitoring framework allows us to record shortcuts or copies from otherwise unmonitored applications.

The current clipboard monitoring mechanism lets us detect when a Windows application is hosting or copying data to the clipboard. The current implementation also records the name of the application from which the data was retrieved. Additionally, the clipboard monitor can receive data in multiple formats. Although most users know that the Windows clipboard can only store a single item, in fact the clipboard can store the same item in multiple formats. For example, after a user copies an image from a Web page, some Web browsers will save the alternate image and text in the snippet memory. In applications that offer a Paste Special on item in the Edit menu in Windows, users can view this option and choose whether to paste a bitmap map or unformatted text.

**6.13.Web usage mining applications**

User behavior is used in various applications such as Personalization, E-commerce, to improve the system and to improve the design of the system according to their interest, etc. ***Web personalization*** offers many features such as simple user greeting to complicate matters further, such as distributing content as per the interests of the users. Content submission is very important as non-expert users are overwhelmed by the amount of information available online. It is possible to predict user behavior by analyzing current navigation patterns with patterns which have been extracted from the online registry. Recommendation systems are the most common application.

Personalized pages are an example of referral systems. E-Commerce applications need customer details for Customer Relationship Management. Mining techniques are very useful to focus on customer attraction, customer retention, cross-selling and customer departure. ***System improvement*** is done by understanding the behavior of Web traffic using mining registry data so that policies can be developed for Internet caching, load balancing, network transmission and data sharing. Patterns for detecting intrusion fraud, breaking attempts are also provided by mining. Performance has been improved to satisfy users. ***Website modification*** is a process of modifying the website and improving the quality of the design and content to recognize the interest of the users. The pages link again according to customer behavior.

**Conclusions**

Websites are one of the most important tools for international advertising for universities and other foundations. The quality of a website can be assessed by analyzing the entries of the website users. To know the quality of a website usage, users need to be evaluated by the site usage mines. Mining results can be used to improve website creation and increase satisfaction that helps in various applications. Registry files are the best source for recognizing user behavior. But raw log files contain unnecessary details like file entries, failed entries, etc, which will affect the accuracy of pattern detection and analysis. So the pre-processing phase is an important job in the mines to do an efficient model analysis. To get accurate mining results the details of the user session need to be known. The survey was conducted on a selection of Internet usage methodologies in the pre-processing proposed by the research community. More focus has been done on the pre-processing stages like session identification and path completion and we have presented different work done by different researchers. Our future research is to create more efficient session reconstructions through graphs and session mines using graph extraction as quality sessions provide more accurate models for user analysis.

The quality of the conclusion in Web mining today is limited by the lack of rich and reliable data. Most of the mined data is read from Web server registers, or Proxy server registers. Unfortunately, the free nature of network transactions on the Web means that little information about the client is transferred to the server. However, by shifting the focus from server to client, data can be collected at source. Much richer data available from the client will enable new observations regarding user behavior. For online retailers, better data could mean more browsers turning into buyers, or more successful sales of related items. For organizations, client-side information can identify which sources of information are most valuable - not because such sources return more results or are more frequently requested, but because those sources contribute to information used elsewhere. As more library services are provided online, librarians can more effectively direct library resources to those areas that are most needed. Client site monitoring represents a significant advancement in server techniques. Most importantly, however, client-side monitoring offers the potential for an integrated analysis of Web usage in a broader usage context.

# **Future Development**

Although Python simplifies development work with Microsoft COM specification, monitoring user events requires some programming skills. Furthermore, even those with prior programming experience should investigate COM events, methods, and properties exposed by a particular application. To ease the burden of implementing the framework, we are investigating the development of a toolkit that would enable non-programmers to select the applications and events they wish to monitor and choose from a menu of actions to take in each case.

For example, a tool user can select the Microsoft Explorer Internet Explorer application, select the "Page loaded" event, and request that the event and related information be saved to a file. The toolkit would generate a NavigateComplete2 "behind the scenes" event handler. This is especially useful for non-technical enterprise librarians or system evaluators who want a graphical control panel to "activate" monitors on an application-by-application basis, with subsequent customized logs and possible automated collection of a group task. Subsequent extensions to the toolkit can support client-side data analysis, based on the most promising analytical techniques.

Our work in the future focuses on four main phases. In the first stage, we will offer a package to enable a simple mechanism to activate and configure client monitoring. In parallel with this phase, we include more client-side productivity tools (such as PowerPoint and regular email clients) in the monitoring framework to increase the range of offers. In the second stage, we will set up automatic communication channels between the individual client or workgroup PCs and an analysis server as mentioned in the previous discussion of workflow support systems. In the third stage, we will perform model analysis on the client activity logs collected on the analysis server and provide a means for the server to feed the findings back to the client interfaces as they interact with the Web and other productivity tools.

Thus, adaptive user interfaces become the means to signal an individual between sessions or a working group between members as knowledge tasks are undertaken. Through such dynamic interfaces, we can provide awareness of the "Workspace", which is the focus of recent CSCW research (Biuk-Aghai & Hawryszkiewycz, 1999; Gutwin & Greenberg, 1998). A fourth stage involves refining our compositional analysis into documents created by individuals or groups within the enterprise. We plan to graphically represent the sources of information that have entered a given version of the document and design in a field setting document compositions, taking into account document genres and working groups. Interesting patterns should be discovered in a longitudinal study using visual and textual component analysis techniques. Taken as a whole, future work requires implementing the client monitoring framework with the widest possible scope and a configuration interface as simple as possible. As server-side analytics and client-adaptive interfaces are implemented, it will be possible to evaluate this framework in a variety of field settings. We look at document composition analysis that has high potential in enterprise knowledge management, as documents are such an important asset of firm knowledge.

There are a number of issues in the pre-processing of registry data. The volume of requests in the Web registry in a single registry file is the first challenge. Analyzing user access log files on the Web helps to understand user behaviors in the structure of the Web to improve the modeling of Web components and Web applications. The registry includes records of document traversal, file recovery, and unsuccessful events on the Web, among many others that are organized according to date and time. It is important to eliminate irrelevant data. So cleaning is done to speed up the analysis as it reduces the number of records and increases the quality of the results in the analysis phase. Attempts in this data to find accurate sessions are likely to be most fruitful in creating highly effective site usage and personalization systems. By following the steps of data preparation, it is much easier to generate rules that identify directories for improving the website. More research can be done in the pre-processing stages to clean up raw registry files, and identify users and build accurate sessions.

**References**

[1] Chaitra L Mugali and Asst. Prof. Padma Dandannavar, “WEB LOGS PRE-PROCESSING AND ANALYSIS: A Survey”, International Journal of Emerging Technology In Computer Science and Electronics, Volume 14, Issue 2, ISSN: 0976-1353

[2] Nanhay Singh, Achin Jain, Ram Shringar Raw , nsingh1973@gmail.com , achin\_jain25@yahoo.com , rsrao08@yahoo.in, Ambedkar Institute of Advanced communication Technologies & Research Delhi, India, “COMPARISON ANALYSIS OF WEB USAGE MINING USING PATTERN RECOGNITION TECHNIQUES”, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.4, July 2013

[3] Mr. Rahul Mishra, Ms. Abha Choubey, Computer Science & CSVTU India, “Discovery of Frequent Patterns from Web Log Data by using FP-Growth algorithm for Web Usage Mining”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 9, September 2012

[4] Doru Tanasa and Brigitte Trousse, AxIS Project Team, INRIA Sophia Antipolis, “Advanced Data Preprocessing for Intersites Web Usage Mining”, 1094-7167/04/$20.00 © 2004 IEEE 59, Published by the IEEE Computer Society

[5] Mr. Shivkumar Khosla, Mrs. Varunakshi Bhojane, Department of Computer Engineering, Mumbai University, India, “Capturing Web Log and Performing Preprocessing of the User’s Accessing Distance Education System”, International Journal of Modern Engineering Research (IJMER) www.ijmer.com Vol.2, Issue.5, Sep.-Oct. 2012 pp-3128-3130 ISSN: 2249-6645

[6] Michal Munk, Jozef Kapusta, Peter Švec, Constantine the Philosopher University in Nitra, Department of Informatics, Tr. A.Hlinku 1, 949 74 Nitra, Slovakia, “Data Preprocessing Evaluation for Web Log Mining: Reconstruction of Activities of a Web Visitor”, International Conference on Computational Science, ICCS 2010

[7] Sheetal A. Raiyani, Shailendra Jain, Dept.of CSE(SS),TIT,Bhopal], “Enhance Preprocessing Technique Distinct User Identification using Web Log Usage data”, ISSN:2249-5789, International Journal of Computer Science & Communication Networks,Vol 2(4), 526-530

[8] Wasvand Chandrama, Prof. P.R.Devale, Prof. Ravindra Murumkar, Department of Information technology, Research scholar of Bharati Vidyapeeth University College of Engineering, Pune, Maharashtra 411046, India., ISSN 2348 –7968, IJISET -International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 10, December 2014.

[9] Abdul Rahaman Wahab Sait, and Dr.T.Meyappan, “Data Preprocessing and Transformation Technique to Generate Pattern from the Web Log”, International conference on Computer Science and Information Systems (ICSIS’2014) Oct 17-18, 2014 Dubai (UAE)

[10] Ramya C, Dr. Shreedhara K S and Kavitha G, M.Tech (Final Year), Professor & Chairman and Lecturer, Dept. of Studies in CS&E, U.B.D.T College of Engineering, Davangere Davangere University, Karnataka, INDIA cramyac@gmail.com and ks\_shreedhara@yahoo.com, “Preprocessing: A Prerequisite for Discovering Patterns in Web Usage Mining Process”, International Conference on Communication and Electronics Information (ICCEI 2011)

[11] V. Sathiyamoorthi, Department of CSE, Sona College of Technology, Salemi-5, and Dr.Murali Bhaskaran, Principal,Paavai College of Engineering, Paachal, 637018, Tamil Nadu, India, “Data Preprocessing Techniques for Pre-Fetching and Caching of Web Data through Proxy Server”, IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.11, November 2011

[12] V. Shanmuga Priya1, S. Sakthivel, Department of computer science, Periyar University, TamilNadu, India, “An Implementation Of Web Personalization Using Web Mining Techniques”, International Journal of Computer Science and Mobile Computing, ISSN 2320–088X, IJCSMC, Vol. 2, Issue. 6, June 2013, pg.145 –150

[13] Gopal Pandey, Swati Patel, Vidhu Singhal, Akshay Kansara, “A Process Oriented Perception of Personalization Techniques in Web Mining”, International Journal of Science and Modern Engineering (IJISME) ISSN: 2319-6386, Volume-1, Issue-2, January 2013

[14] Surbhi Anand, Surbhi Anand, Department of Computer Science & Engineering, Thapar University, Patiala-147004 (India), “An Efficient Algorithm for Data Cleaning of Log File using File Extensions”, International Journal of Computer Applications (0975 –888) Volume 48–No.8, June 2012

[15] Ankit R Kharwar1, Chandni A Naik2, Niyanta K Desai3, 1Assistant Professor, Department of Computer, Chhotubhai Gopalbhai Patel Institute of Technology, Bardoli , 2,3Student of M.Tech Computer Engineering in Chhotubhai Gopalbhai Patel Institute of Technology, Bardoli, “AComplete Pre-Processing Method for Web Usage Mining”, International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 10, October 2013)

[16] Ms. Dipa Dixit Lecturer, Ms. M Kiruthika Assistant Professor, Fr.CRIT, Vashi, “Preprocessing Of Web Logs”, (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 07, 2010, 2447-2452

[17] Sheetal A. Raiyani, Shailendra jain, “Efficient Preprocessing technique using Web log mining”, 1Department of CSE(Software System),Technocrats Institute of Technology, Bhopal,India; 2Department of CSE, Technocrats Institute of Technology, Bhopal,India, International Journal of Advancements in Research & Technology, Volume 1, Issue6, November-2012 1 ISSN 2278-7763

[18] Vijayashri Losarwar, Dr. Madhuri Joshi, “Data Preprocessing in Web Usage Mining”, International Conference on Artificial Intelligence and Embedded Systems (ICAIES'2012) July 15-16, 2012 Singapore

[19] Nanhay Singh, Arvind Panwar and Ram Shrinagar Rao, Ambedkar Institute of Advanced Communications Technologies and Research, Delhi, India: “Enhancing the Performance of Web Proxy Server through Cluster Based Pre-fetching Techniques”, International Conference on Advances in Computing, Communications and Informatics (ICACCA), 2013

[20] R. Lokeshkumar, R. Sindhuja, Dr. P. Senguttuvelan, Assistant Professor –(Sr.G), PG Scholar, 3Associate Professor, Department of Information Technology, Bannari Amman Institute of Technology, Tamilnadu: “A Survey on Preprocessing of Web Log File in Web Usage Mining to Improve the Quality of Data”, International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume-4, Issue-8, August 2014

[21] Naga Lakshmi, Raja sekhar Rao, Sai Satyanarayana Reddy: “An Overview of Preprocessing on Web Log Data for Web Usage Analysis”, Published in International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-2, Issue-4, March 2013

[22] Mehak(ME), Mukesh Kumar (Assistant Professor), Naveen Aggarwal (Assistant Professor), Computer Science & Engineering Department, University Institute of Engineering &Technology, Panjab University, Chandigarh, India, “Web Usage Mining: An Analysis”, Journal Of Emerging Technologies In Web Intelligence, Vol. 5, No. 3, August 2013