IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. VIII (Mar-Apr. 2014), PP 63-71
www.iosrjournals.org

"Mining the web data using data mining techniques for identifying and classifying the user access behavioral patterns"

Shruthi C Kamoji[1] , Praveen Naik[2]
[1] *Computer science & Engineering, Acharya Institute Of Technology, Karnataka, India*
[2]*Assistant Professor, Computer science & Engineering, Acharya Institute Of Technology, Karnataka, India*

Abstract: *The one of the largest and most widely used document repository is worldwide web. It has been used for mining data since many decades. It has been proved as one of the most helping platform to assimilate, disseminate and retrieve information. But unfortunately its success has only become its enemy. It seems like an ocean of information in which users are drowning not sailing. The information is so huge, diverse, dynamic and unstructured natured that users face the problems of information overloaded while interacting with the web. Here the issue of QoS cops up. It's needed for web developer to know what the user really wants to do, predict which pages the user is interested in and provide the user the WebPages by knowledge of users navigational patterns to improve QoS. This project mainly focuses on cleaning the data i.e. sever web log file, processing the data according to some specific strategy, identifying the users using maximal forward reference algorithms and classifying them into predefined classes. Here supervised learning is used to train the classifier. We have carried out this project using an educational institute's log file as input data.*
Our project work has been used to implement the model for providing the desired information to the user if the data at the backend can be maintained appropriately and grouped into different classes. We have done the implementation using the corresponding data can be given to the user, ie instead of giving hundreds of links related topic to the user, more appropriate links pertaining to the user can be given. Hence the prevision rate can be increased.

Keywords*: QOS, WEBMINER, clustering, Session, GiniIndex, Raw web log data*

I. World Wide Web

` Internet is used to access the system of interlinked hypertext documents called as the **World Wide Web** (abbreviated as **WWW** or **W3**, and commonly known as **the Web**). Web pages may comprise of text, images, videos and other multimedia. one can view web pages through a web browser and navigate between them via hyperlinks. The World Wide Web had a many more differences from other hypertext systems that were available at that time. The Web needed only unidirectional links rather than bidirectional ones. This made it possible for someone to link to another resource without action by the owner of that resource. It also significantly reduced the difficulty of implementing web servers and browsers (in comparison to earlier systems).

II. Data Mining

A relatively young and interdisciplinary field of computer science known as **Data mining** which is the analysis step of the knowledge discovery in databases process, or KDD, is the process of discovering new patterns from large data sets involving methods at the intersection of machine learning, artificial intelligence, statistics and database systems. The overall goal of the data mining process[1] is to extract knowledge from a data set in a human-perception structure and besides the raw analysis step involves database and data management aspects, data preprocessing, model and inference considerations, interestingness metrics, complexity considerations, post-processing of found structure, visualization and online updating

III. Web Mining

This technology has many vital roles that are worth mentioning. It can automatically find, extract information from the variety web resources. It also develops, enhances and improves the quality and the efficiency of search engines, makes classifications determines web pages or files. It can also generate large-scale real-time data. Web data mining [2] discovers useful information from the Web hyperlink and page content. These features are new in web mining and they have already changed the status of many business functions in a modern competitive enterprise. It becomes easier to make correct business decisions or perceive the information that came from customers with the help of web data mining.

IV. Web Usage Mining

Web usage mining is the process of extracting useful information from server logs i.e the history of users using the web. Web usage mining is the process of finding out what users are looking for on the Internet, what users are exactly interested in. Some users might be looking at multimedia data, whereas some others might be interested only textual data.

4.1 Architecture of web mining

The WEBMINER is a system that implements parts of this general architecture. The architecture divides the Web usage mining process into two main parts. The first one includes transforming the Web data[6] into suitable transaction form which is the domain dependent processes. This includes preprocessing, transaction identification, a nd data integration components. The second part includes the largely domain independent application of generic data mining and pattern matching techniques (such as the discovery of association rule and sequential patterns) as part of the system's data mining engine. The overall architecture for the Web mining process is depicted in Figure.1.

Data cleaning is the first task to be performed in the Web usage mining process. After the cleaning of data, the log entries must be partitioned into logical clusters using one or a series of transaction identification modules. The clean server log can be thought of in two ways; either as a single transaction of many page references, or a set of many transactions each consisting of a single page reference. The task of identifying transactions is one of either merging small transactions into fewer larger ones or dividing a large transaction into multiple smaller ones.

Once the domain-dependent data transformation phase is completed, the resulting transaction data must be formatted to conform to the data model of the appropriate data mining task. For instance, the format of the data for the association rule discovery task may be different than the format necessary for mining sequential patterns

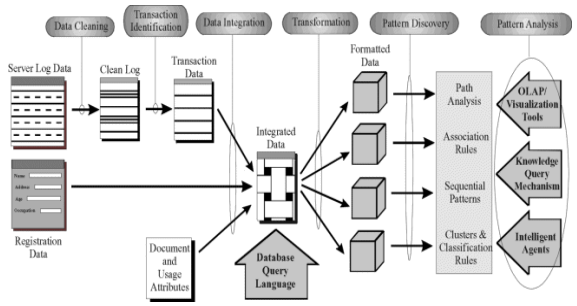

Fig -1: general architecture of web mining process

V. Problem Definition

5.1 Current difficulties:

The World Wide Web is very widely used and popular amongst all sectors of the people. As its usage is increasing rapidly, the prominent problem are emerging first, managing a very huge database is becoming complex as information on web is very volatile and large. Second problem is that, the large number of websites is being developed in higher speed they sometimes fail to provide relevant data or exact data as demanded by the user. So to overcome these two problems many are carrying out research in web mining area. In these aspects identification of users has become tedious task which in turn mislead the classification of that user.

5.2Scope and Objective:

- To efficiently and effectively identify the users on the web based on their content and usage.
- To identify sessions based on a particular criteria.
- To classify the users based on their interests, web contents.

VI. DESIGN ASPECTS

The purpose of data preprocessing is to extract useful data from raw web log and then transform these data in to the form necessary for access pattern analysis. Here we are using web log file of an educational institution.

According to the given phases the data is preprocessed as follows:

Raw Web Log Data:. The purpose of data preprocessing is to extract useful data from raw web log and then transform these data in to the form necessary for pattern discovery. Due to large amount of irrelevant information in the web log, the original log cannot be directly used in the web log mining procedure, hence in data preprocessing phase, raw Web logs need to be cleaned, analyzed and converted for further step. The data recorded in server logs, such as the user IP address, browser, viewing time, etc, are available to identify users

and sessions. However, because some page views may be cached by the user browser or by a proxy server, we should know that the data collected by server logs are not entirely reliable.

Data Cleaning: Data cleaning means eliminate the irrelevant information from the original Web log file Usually, this process removes images, multimedia files, and page style files requests concerning non-analyzed resources. For example, requests for graphical page content (*.jpg & *.gif images) and requests for any other file which might be included into a web page or even navigation sessions performed by robots and web spiders. When the useless data is filtered, log file size is reduced to use less storage space and to facilitate upcoming tasks

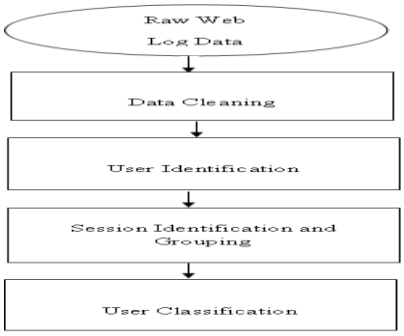

Fig.-2 Flow Diagram of the project

Session Identification: Session identification is dividing the pages accessed by each user into individual session. The goal of session identification is to find each user's access pattern and the paths he is accessing frequently. The simplest method is using a timeout, where if the time between page requests exceeds a certain limit, it is assumed that the user is starting a default timeout. Once a site log has been analyzed and usage statistics obtained, a timeout that is appropriate for the specific Web site can be fed back in to the session identification algorithm.

Grouping of similar sessions and classification: In this phase we deal with the sessions that are defined and classifying them by grouping users of similar sessions. By grouping the sessions we get a structured format to our database so that it can be accessed more efficiently. As we are providing a front end to identify the users in a particular session according to the given log file, we can compare the efficiency of our algorithm pertaining to the previously proposed algorithms, with the help of graphs.

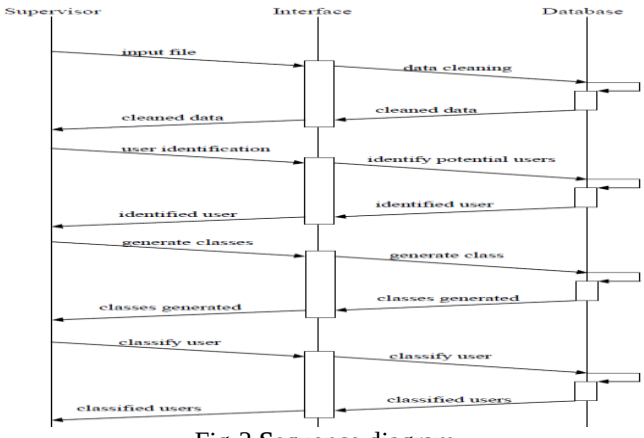

Fig-3 **S**equence diagram

VII. Raw Web Log Data

The primary data source used in web mining is Log file. These are obtained mainly from server, this may also include web access log files and also can be obtained by client- server or proxy server. These log files which are obtained from the various sources become the raw data for the web usage mining. We have collected raw web log data from an institution on daily basis. These log files usually contain date & time, ip address (client), url accessed, status code, status description. Url status code is a 3 digit code usually describing the status of the url accessed. Status description is usually hit or miss. Url specifies the link accessed by the client. The general format of the log table is given in table below.

Field	Date	Description
Date	date	The date that the activity occurred
Time	time	The time that the activity occurred
Client IP address	c-ip	The IP address of the client that accessed your server
User Name	cs-username	The name of the autheticated user who access your server, anonymous users are represented by -
Servis Name	s-sitename	The Internet service and instance number that was accessed by a client
Server Name	s-computername	The name of the server on which the log entry was generated
Server IP Address	s-ip	The IP address of the server that accessed your server
Server Port	s-port	The port number the client is connected to
Method	cs-method	The action the client was trying to perform
URI Stem	cs-uri-stem	The resource accessed
URI Query	cs-uri-query	The query, if any, the client was trying to perform
Protocol Status	sc-status	The status of the action, in HTTP or FTP terms
Win32 Status	sc-win32-status	The status of the action, in terms used by Microsoft Windows
Bytes Sent	sc-bytes	The number of bytes sent by the server
Bytes Received	cs-bytes	The number of bytes received by the server
Time Taken	time-taken	The duration of time, in milliseconds, that the action consumed
Protocol Version	cs-version	The protocol (HTTP, FTP) version used by the client
Host	cs-host	Display the content of the host header
User Agent	cs(User Agent)	The browser used on the client
Cookie	cs(Cookie)	The content of the cookie sent or received, if any
Referrer	cs(Referrer)	The previous site visited by the user. This site provided a link to the current site

Fig-4 General format of log table

Some servers follow extended log format along with referrer and user agent. User agent is the string describing the type and version of browser software used Referrer is the referring link url. The two drawbacks are web cache and the IP address misinterpretation in the server log. Web cache keeps track of web pages that requests and saves a copy of these pages for a certain period. If there is a request for same page, the cache page is in use instead of making new request to the server. Therefore, these requests are not record into the log files. Here is the example of the web log file.

```
192.168.255.155, anonymous, 08/20/2009, 7:55:20, WMW, SALES1,
192.168.114.201, 4502, 163, 3223, 200, 0, POST, /a.htm, —,
192.168.255.154, anonymous, 08/20/2009, 7:55:21, WMW, SALES1,
192.168.114.201, 4502, 163, 3223, 200, 0, POST, /a.htm, —,
```

Fig-5 Eg of web log file

VIII. Data Cleaning

Data cleaning refers to removal of the irrelevant information from the original Web log file. There are two kinds of irrelevant or redundant data needed to be cleaned. They are accessorial resources embedded in HTML file and error requests.

1) Accessorial Resources. Because HTTP protocol is connectionless, a user's request to view a particular page often results in several log entries since graphics and scripts are down-loaded in addition to the HTML file. Since the main motto of Web Usage Mining is to get a picture of the user's behavior, inclusion of file requests that the user did not explicitly request does not make any sense. Elimination of the items that seems irrelevant can be reasonably accomplished by checking the suffix of the URL name. For instance, all log entries with filename suffixes such as gif, jpeg, GIF, JPEG, jpg, JPG, css and map can be removed. In addition, common scripts such as the files requested with the suffixes of ".cgi" can also be removed.

2) Error's requests: Error's requests are not useful for mining process. They can be removed by checking the status of request. Each time you enter a url server returns a status code. There are four types of status code returned by server they are: Redirect (300 Series), Failure (400 Series), Server Error (500 Series) & Success (200 Series) . The range for hit is 200 to 299. In this phase, entries with only GET and POST method are retained and others are filtered out.

After removing the data that is irrelevant from the web log file the obtained file is then stored in to a data base for further access. Once the cleaned data is obtained the next step is assigning a unique id for all unique url present in the data base. It reduces complications to identify specific urls. The proposed algorithm for Data cleaning is as follows.

ALGORITHM: Data Cleaning
Input: Web Server Log File.
Output: Log Database.
Step1: Read Log Record from Web Server Log File.
Step2: If status code <200 or >= 299, discard the record.
Step3: Url is checked.
Step4: If the file is image file like .gif, .jpg, .jpeg etc discard the record.
Step4: Convert the milliseconds into date and time format.
Step5: Go to step 1.

Step6: If end of file is reached stop the process.

The outcome of this algorithm is the LogDatafile consisting relevant set of records with the entries as user name, ip address, Date, Time, and url details etc. After data has been cleaned most simultaneously repeated entries should be removed so as to avoid the repetitions of the entries, to do so here is a method:

First find the url which is repeated simultaneously having the same ip address. Then check the time limit between the two records, if the time interval is less than 20 minutes the second record is neglected and only the first record in the web log file is considered. By doing so the URL repetitions can be avoided easily .Once the cleaned log file without having any repetitions is obtained, this log file is been written into the database for further reference. After the log file is entered into the data base the next step is assigning an unique identification number called urlid. For every unique url entry in the log data base an unique identification number is assigned. Here in our project we have assigned a character followed by an integer value .

After Data Cleaning Phase the result can be shown as:

Raw Web Log	Cleaned Data	Efficiency
3.05 GB	689 KB	99.97%

Fig-6 After Data Cleaning Phase

IX. User & Session Identification:

User identification is the crucial step in data preprocessing model. Since its very challenging task to find out a particular user from heaps of the log records stored in the server, there are many ways to identify the users. The areas are as follows:

a. **Software agents**

These are small application modules which are installed in user computer. This keeps track of all the web transactions of the user. The only assumption is that both the server and user have the same application and information.

b. **User ID**

Here identifying is accurate since users themselves give their identification data through user id and passwords. Here the only assumption is all servers provide the registration forms.

c. **Cookies**

These are the small pieces of information stored in user's computer by the server. It is most efficient technique but the user must have set the cookie on his machine otherwise no information is stored.

d. **Proxy server**

Enhanced proxy servers can also provide reasonably accurate user identification However, they have many disadvantages. They require that the user register their computer with a proxy server.

e. **Session ID's**

When user revisits the same site from the same computer, the same userid is used. There is no burden placed on the user at all. However, if the user uses more than one computer, each location will have a separate cookie, and thus a separate user profile. Also, if the computer is used by more than one user, and all users share the same local user id, they will all share the same, inaccurate profile.

Sessions are normally defined as the no of page visits done by the user in a certain time allotment. It depends on the particular user whether he/she has single or multiple sessions. Here even we can redefine user's access pattern in that particular session by using some reconstruction techniques.

The identification of the users is done primarily on the basis of ip address. We've implemented the Maximal Forward Reference algorithm for identifying users. The simple representation of the algorithm can be given as

IP	Accessed Url	
192.168.1.13	A1	
192.168.1.13	A2	
192.168.1.13	A3	USER 1
192.168.1.13	A4	
192.168.1.13	A5	
192.168.1.13	A1	
192.168.1.13	A6	
192.168.1.13	A7	USER 2
192.168.1.13	A8	
192.168.1.13	A9	

Fig-7 Representation of maximal forward reference

In this algorithm user pattern is analysed and user is identified if the user makes a backward reference to the url with which he had started browsing. The following is the rules we use to identify user session in our experiment along with maximal forward reference concept:

1) If there is a new user, there is a new session;
2) In one user session, if the refer page is null, there is a new session;
3) If the time between page requests exceeds a certain limit (30 or 25.5mintes), it is assumed that the user is starting a new session.

The algorithm used for user identification is as follows:

ALGORITHM: User Identification

Input: Log database.

Step1: The ip address is checked.

Step2: If different ip address is accessed
then assign as different user.

Step3: If same ip accessed is
then pattern of user access is examined.
Check for any backward reference of the url.

Step4: If (any backward reference of url)
then group the records from first url until repetition of url .

Step5: If ((repeated url)!=(first record))
then group all the previous urls and proceed.

Step6: Consider time limit=30 min for grouping.

Step7: Go to step 1 and repeat the process until EOF.

Step8: Stop the process.

After identifying potential users, we have used at particular user's access pattern. We have taken the host name of the url which user has accessed and made different sessions of single user.

X. Classification

In classification a data item is mapped into one of several predefined classes. In the internet marketing, a customer can be classified as 'no customer', 'visitor once' and 'visitor regular' based on their browsing patters and discovered rules for attracting the customers by displaying special offers

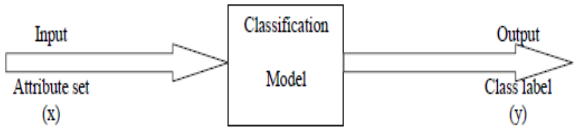

Fig-8 Classification as the task of mapping an input attribute x in its class label y

In the web domain, one is interested in developing a profile of users belonging to a particular class or category. This requires extraction and selection of characteristics that best describe the properties of a given class or category. Supervised inductive learning algorithms such as decision tree classifiers, naïve Baysian classifiers, k-nearest neighbor classifiers, Support Vector Machines etc can be used to the classification. There are two types of learning methodology:

Supervised learning (classification):

The training data (observations measurements, etc.) are accompanied by labels indicating the class of the observations. Here new data is classified based on the training set.

In our project we have used six predefined classes.

Predefined classes
Entertainment
Education
Sports
Jobs
Research
Ecommerce

Fig-9 Predefine classes

Unsupervised learning (clustering):

The class labels of training data are unknown. Given a set of measurements, observations, etc with the aim of establishing the existence of classes or clusters in the data.

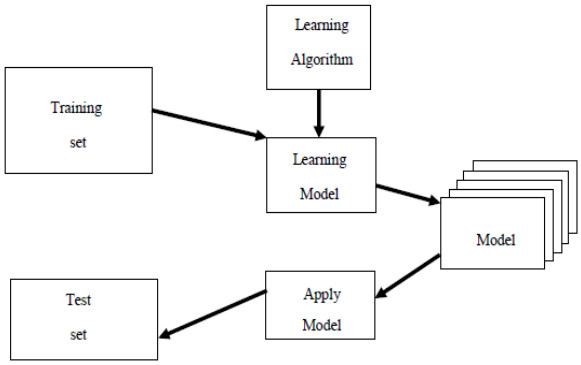

Fig-10 General approach for building classification model

10.1 Gini index (ibm intelligentminer)

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

If a data set T contains examples from n classes, gini index, gini(T) is defined as where p_j is the relative frequency of class j in T.

If a data set T is split into two subsets T_1 and T_2 with sizes N_1 and N_2 respectively, the gini index of the split data contains examples from n classes, the gini index gini (T) is defined as

$$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$ sen to split the node (need to enumerate all possible splitting points for each attribute).

As we are using supervised learning this feature is not used in our project. This feature is very essential while clustering process whenever there is a confusion to split a data when it features relates to more than one class we use gini split to determine which class it belongs more

Let the set of examples S contain p elements of class P and n elements of class N. The amount of information, needed to decide if an arbitrary example in S belongs to P or N is defined as

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

For eg:

07/02/2012 15:18:45 0 192.168.0.90 TCP_HIT/205 1726 GET http://gmail.com/ - NONE/- text/html
07/02/2012 15:25:43 0 192.168.0.90 TCP_HIT/205 1404 GET http://rospres.com/ - NONE/- text/html
07/02/2012 15:38:36 0 192.168.0.162 TCP_HIT/205 1876 POST http://IPLlive.com - NONE/- text/html

07/02/2012 15:44:19 0 192.168.0.162 TCP_HIT/205 1876 POST http://microsoft.com - NONE/- text/html

In this data, first two records corresponds to 192.168.0.90 ip address and next two records are corresponds to 192.168.0.162 ip address. Now we can observe that even if first two records have same ip address, they differ in timing. A same thing applies to next two records. By this we can identify that ip address can't be used as an attributes to classify the identified users as follows. If we consider 1st record as user 1, 2nd record as user 2, 3rd record as user 3, and 4th record as user 4 using time constraints, then according to the url's accessed by user 1, U1 is classified as entertainment and so on.

Another eg applying the gini index for test data including 10 records:

07/02/2012 15:38:28 0 192.168.0.90 TCP_HIT/205 1726 GET http://gmail.com/ - NONE/- text/html
07/02/2012 15:39:45 0 192.168.0.90 TCP_HIT/205 1404 GET http://facebook.com/ - NONE/- text/html
07/02/2012 15:42:24 0 192.168.0.90 TCP_HIT/205 1876 POST http://youtube.com - NONE/- text/html
07/02/2012 15:43:25 0 192.168.0.90 TCP_HIT/205 1876 POST http://gmail.com - NONE/- text/html
07/02/2012 15:44:56 0 192.168.0.90 TCP_HIT/205 1726 GET http://youtube.com/ - NONE/- text/html
07/02/2012 15:46:06 0 192.168.0.90 TCP_HIT/205 1404 GET http://rospres.com/ - NONE/- text/html
07/02/2012 15:47:36 0 192.168.0.90 TCP_HIT/205 1876 POST http://microsoft.com - NONE/- text/html
07/02/2012 15:49:55 0 192.168.0.90 TCP_HIT/205 1876 POST http://enggresource.com- NONE/-text/html
07/02/2012 15:51:23 0 192.168.0.90 TCP_HIT/205 1726 GET http://vtu.ac.in/ - NONE/- text/html
07/02/2012 15:53:34 0 192.168.0.90 TCP_HIT/205 1404 GET http://ieee.com/ - NONE/- text/html

There are totally 10 records in test data. We know that there are two kinds of users here i.e. entertainment-oriented (P) and education-oriented (N).

Case 1: If we consider IP address, we have 10 records of same ip (say) p=10, n=0 and p+n=10+0=10. So according to the equation we get:

$I(p,n)= -\frac{10}{10}\ log_2\ \left(\frac{10}{10}\right) - \frac{0}{10}\ log_2\ \left(\frac{0}{10}\right)$

I(p,n)= 0, thus no information gain.

Case 2: Let us now consider urls. By the host name of the urls accessed we can say that first five belongs to entertainment i.e. gmail, facebook & youtube. The next four belongs to education i.e. rospres, Microsoft, enggresource & vtu. The last record is unknown. Thus, p=5, n=4 and p+n=5+4=9.

$I(p,n)= -\frac{5}{9}\ log_2\ \left(\frac{5}{9}\right) - \frac{4}{9}\ log_2\ \left(\frac{4}{9}\right)$

I(p,n)=0.9910 i.e. 99.10%.

By applying Gini Index we have come to know that, among the potential attributes i.e. date, time, ip address and urls, case 2 has the highest information gain through which we can extract access behavior of a particular user. In this project date and ip address is used for identifying a user uniquely in addition with the urls. Thus, we are selecting the attribute url to classify the users after identifying them.

10.2 Decision tree

Decision tree is a classifier in the form of a tree structure

- **Decision node**: specifies a test on a single attribute.
- **Leaf node**: indicates the value of the target attribute.
- **Arc/edge**: split of one attribute.
- **Path**: a disjunction of test to make the final decision.

In a decision tree, class label is assigned to each leaf node. The non terminal nodes, which include root nodes and internal nodes, contain attribute test conditions to separate records which have different characteristics. Starting from the root node; we apply the test condition to the record and follow the appropriate branch bound on the outcome of the test. This will lead us either to another internal node, for which a new test condition is applied, or to a leaf node. The class label is associated with the node is then assigned to the record.

Decision trees classify instances or examples by starting at the root of the tree and moving through it until a leaf node. Decision trees are powerful and popular tools for classification and prediction. Decision trees represent rules, which can be understood by humans and used in knowledge system such as database.

Requirements of a decision tree:

- **Attribute-value description:** object or case must be expressible in terms of a fixed collection of properties or attributes (e.g., hot, mild, cold).
- **Predefined classes (target values):** the target function has **discrete output values** (bollean or multiclass)
- **Sufficient data:** enough training cases should be provided to learn the model.

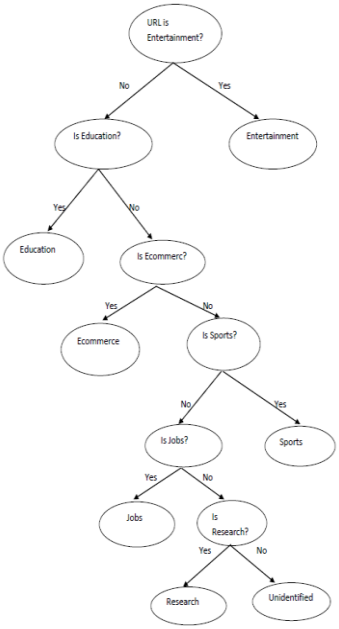

Fig-11 decision tree

XI. Conclusion

The word WEB has evolved rapidly over the last few years. Despite several researches carried out in web mining over several years, effective extraction of the relevant information from the web is still the challenging task. So there is a need for the user identification based on their access pattern, which can be solved by identification and classification of the users into predefined classes by means of web usage mining. The machine generated web log [5]file is processed and then users are classified in a well defined manner.

We have implementation maximal forward reference which is not used in any of the web mining field. We have even used gini index for the attribute selection. Web usage mining is all about extracting the useful information from the web, processing it as per the user requirements. In this project we have mainly concentrated on identifying the users using maximal forward reference method.

References

[1] S.Vijayalakshmi, V.Mohan, "MINING OF USERS'ACCESS BEHAVIOUR FOR FREQUENT SEQUENTIAL PATTERN FROM WEB LOGS", International Journal of Database Management Systems (IJDMS) Vol.2, No.3, August 2010

[2] Wei Gao, Olivia R. Liu Sheng" Minin Characteristic Patterns to Identify WebUsers", 2006

[3] Maria J. Martín-Bautista, María-Amparo Vila, Victor H. Escobar-Jeria. "OBTAINING USER PROFILES VIA WEB USAGE MINING". IADIS European Conference Data Mining 2008.

[4] Uichin Lee, Zhenyu Liu, Junghoo Cho "Automatic Identification of User Goals in Web Search", University of California Los Angeles, CA 90095, 2010.

[5] Ford Lumban Gaol,"Exploring The Pattern of Habits of Users Using Web Log Squential Pattern" 2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies.

[6] V.Chitraa, Dr. Antony Selvdoss Davamani, "A Survey on Preprocessing Methods for Web Usage Data". (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, No. 3, 2010.

[7] Li Chaofeng, "Research and Development of Data Preprocessing in Web Usage Mining". School of Management, South-Central University for Nationalities, Wuhan 430074, P.R. China, 2010.

[8] K.R.Suneetha, R. Krishnamoorti, "IRS: Intelligent Recommendation System for Web Personalization", European Journal of Scientific Research, Inc. 2011