The Libyan Academy of Graduate Studies
Misrata

School of Applied Science

Department of Computer Science

Applying Web Mining Application for Understanding User's Behavior

Prepared by:

Musab Saleh Alriani

Supervised by:

Prof. Dr. Zakaria Suliman Zubi

Fall 2014

Acknowledgment

In the name of Allah, the most merciful, the most compassionate all praise be to Allah, the lord of the worlds; and prayers and peace be upon Mohamed his servant and messenger the first and the foremost. I must acknowledge my limitless thanks to Allah, the Ever-Magnificent; the Ever-Thankful, for his helps and bless. I am totally sure that this work would have never become truth, without his guidance, also I would like to express my gratitude to my supervisor, Prof. Zakaria Suliman Zubi, whose expertise, understanding, patience, added considerably to my graduate experiences.

I appreciate his vast knowledge and skills in many areas of computer science, and his assistance in writing my proposals and thesis, I owe a deep debt of gratitude to the school's administration of applied science, the department of computer science and the employees of the school for their dedication to provide student of this great facility the comfort and the necessary needs tools to success.

Dedication

This thesis is dedicated to: The sake of Allah, my Creator and my Master;

My great teacher and messenger, Mohammed (May Allah bless and grant him), who taught us the purpose of life;

My homeland Libya;

My passed father, who taught me three facts about knowledge are determining goals, discipline, and working hard ,all of that pays off eventually.

It is also dedicated to my great mother, who taught me that patience is the key of all good that can accomplish the largest task if it is done one step at a time.

My brothers and sisters for their care and support.

At last not at least my dear friends for help and encouragement.

Abstract

The Internet is widely known as the biggest source of growing information ,it consisted of millions of documents called web pages, to access these web pages by the user of internet, he or she depends on the browser to connect to the machine called web server where data is stored, by sending requests to web sever for the required web pages returned in HTML format loading out the text and graphics on the user's computer screen, all the previous transactions occurred between the web server and the browser are recorded in a file named web log file exists in a web server.

Web log file is an archive information about the history of requested web pages , in addition to request date/time ,client ip address, user agent (the OS used) and referrer(the last pages the user was on),in this thesis these information are used to extract useful knowledge to provide a better understanding of users behavior.

First data in log file is prepared the for processing (preprocessing stage),by selecting the necessary records e.g. client ip address, date/time record and requested pages, also removing the unnecessary data e.g. requested page leads to view or download picture or video ends with extension like jpg,gif,wam,kmp , from the extracted records , the sessions for every user are identified (session is a set of pages the user has visited in specific time) to form the transaction database ,in the next stage data mining techniques are applied to transaction database (pattern discovery) ,such as association rules techniques to describe the user pattern through the correlated visited pages ,also classification techniques are used to classify every page from the set of correlated pages to its category, then the classification results are analyzed , using the visualization techniques(pattern analysis) by presenting the classification result for every user in graphical manner clarifies the user behavior showing the differences between all users behaviors.

Table of Contents

Chapter 4: Implementation

Chapter 5: Results and Conclusions

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

Chapter 1

1.1 Introduction

The Internet offers a huge, widely global information center for news, advertising, consumes information, financial management, education, government, and e-commerce. It contains rich and vital information, about the contents of a web page with hyperlink structures and multimedia, and hyperlink information.

The access and use of information provides fertile sources for extracting data. Consequently, web mining is the application of data mining techniques to discover patterns, and knowledge from the web. Web mining is the application of data mining techniques to extract knowledge from web data, including Web documents, hyperlinks between documents, usage logs of web sites [1],the aim of using web mining techniques for understanding user behavior is to profile user characteristics. Web mining works as an inductive data analysis which produces various interesting and valuable patterns from the collected data. Large amount of data is produced from websites data which can reveal conditions for example of whether a user prefers to visit a specific type of sites or not.

according to analysis targets, web mining can be organized into three main categories as shown in figure 1.1 below:

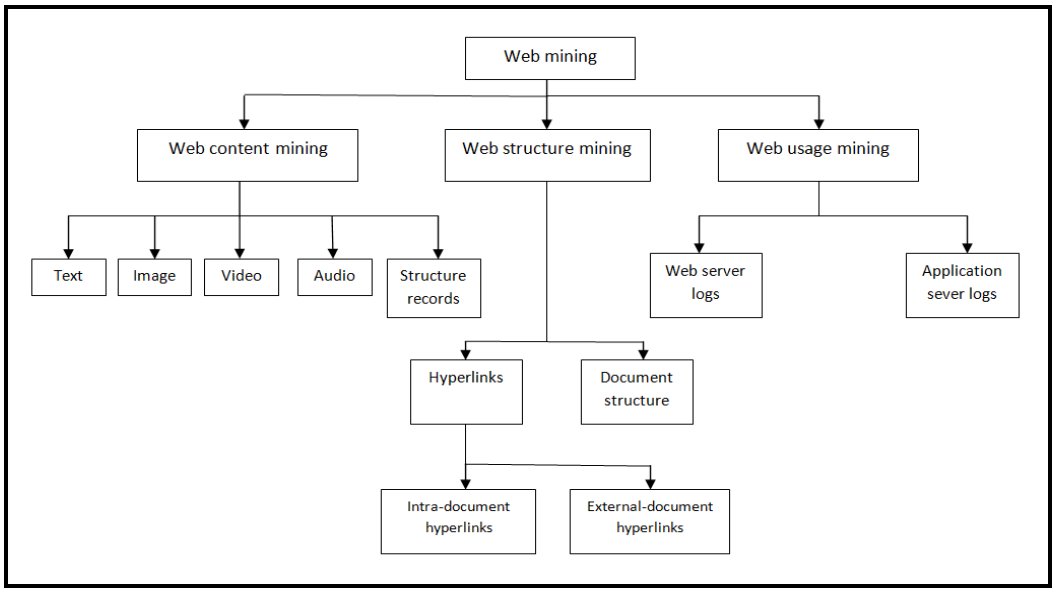


Figure 1.1: web mining categories and their elements

1-Web Content Mining:

Web Content Mining is the process of extracting useful information from the contents of Web documents. Content data corresponds to the collection of facts a Web page was designed to convey to the users ,It may consist of text, images, audio, video, or structured records such as lists and tables[2].

The form of web content mining data is divided in two types

- Unstructured data such as free text .
- Structured data as HTML documents and more structured data as tables and data base generated HTML pages.

Application of text mining to Web content has been the most widely researched.

Web content mining analyzes web content data by applying data mining techniques such as extracting association patterns, clustering of web documents and classification of Web Pages.

The goal of web content mining is mainly to improve information, finding or filtering Information, building a new model of data on the web to improve the searching process, by analyzing web content such as text, multimedia data, and structured data (within web pages or linked across web pages)[3].

2-Web structure mining:

Web structure mining is the process of using graph and network mining theory and methods to analyze the nodes and connection structures on the web [4].

These are as follows:

- *Hyperlink*: It is an element in an electronic document that links to another place in the same document or to an entirely different document. Hyperlinks are divided into two types.
  1- Internal-document hyperlinks that lead to pages within the same website.
  2- External -document hyperlinks that lead to other web pages.
- Document Structure: It is a schema language for XML that is a language for describing valid XML documents [5].

3-Web usage mining

web usage mining is the application of data mining techniques to discover interesting usage patterns from web usage data in order to understand and better serve the needs of web based application[6], It emphasizes on the knowledge discovered while the user is navigating through the websites. That means all kinds of user requests and maintaining a repository of all such requests in log files. Web usage mining is classified into two types.

- Web Server Data:

Logs are made by the web server and they include field like IP addresses, the web pages accessed and the access times.

- Application Server Data:

Such applications are prepared for carrying out the business transactions and make their repository in application server logs [6].

Web usage mining is a process explained briefly as shown in figure 1.2 divided in three stages of data mining cycle, including data preprocessing, pattern discovery & pattern analysis [7].

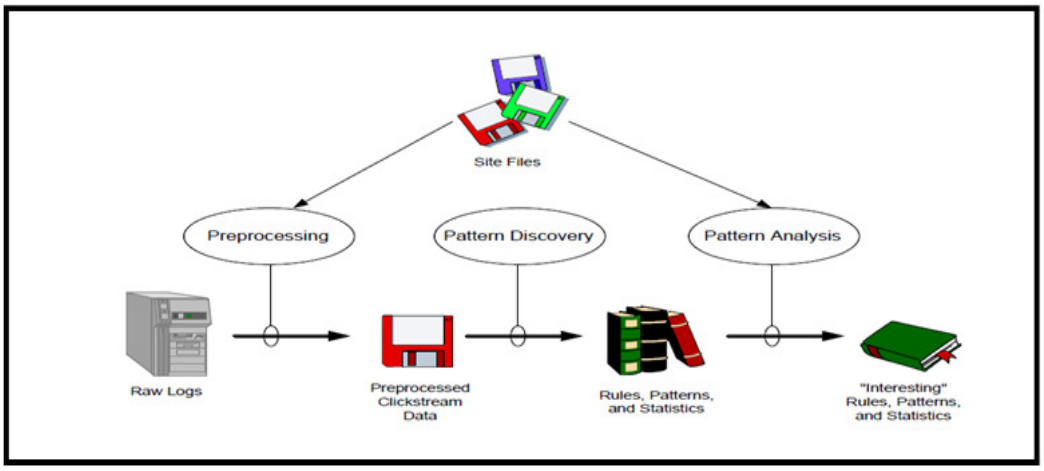
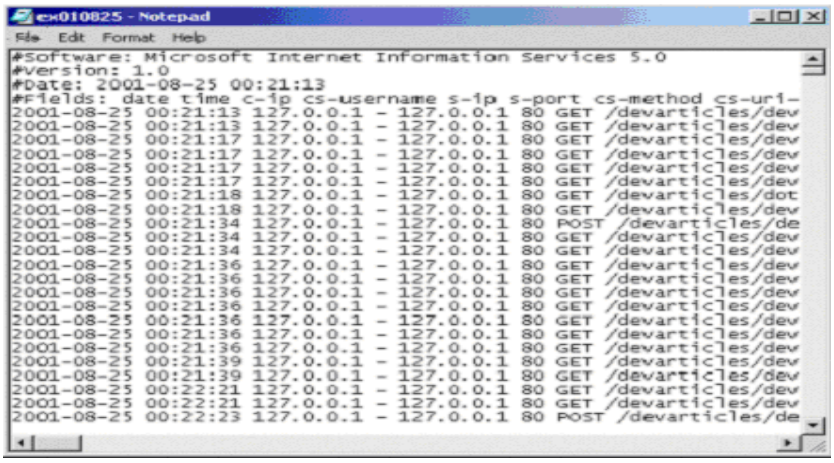


Figure 1.2: the web usage mining process

1.2 Field of study

Among web mining branches ,this thesis focuses on web usage mining which is about user browsing activates in the web leaving a massive amount of data occurred because of transaction between user and server recorded on text file as shown in figure 1.3 called web log file.


ex010825 - Notepad

File Edit Format Help

```
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2001-08-25 00:21:13
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-
2001-08-25 00:21:13 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:13 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:17 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:17 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:17 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:17 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:18 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dot
2001-08-25 00:21:18 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:34 127.0.0.1 - 127.0.0.1 80 POST /devarticles/de
2001-08-25 00:21:34 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:34 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:36 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:39 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:21:39 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:22:21 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:22:21 127.0.0.1 - 127.0.0.1 80 GET /devarticles/dev
2001-08-25 00:22:23 127.0.0.1 - 127.0.0.1 80 POST /devarticles/de
```

Figure 1.3: Web Log File in Text format.

It contains information about request entries includes date,time, c-ip, cs-username, s-ip, s-port,cs-method,cs-uri-stem,cs-uri-query,sc-status, sc-bytes, cs-bytes time-taken cs(User-Agent) cs(Referrer),all those entries are explained in the following example as mentioned in the figure 1.4 and table 1.1

```
#Software: Microsoft Internet Information Services 6.0

#Version: 1.0

#Date: 2002-05-24 20:18:01

#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-
query sc-status sc-bytes cs-bytes time-taken cs(User-Agent) cs(Referrer)

2002-05-24 20:18:01 172.224.24.114 - 206.73.118.24 80 GET /Default.htm - 200 7930
248 31 Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+2000+Server)
http://64.224.24.114/
```

Figure 1.4 Sample web log file entry values.

You The preceding log file entry values can be interpreted as follows :
#Software: Microsoft Internet Information Services 6.0: This field indicates the version of IIS that is running.
#Version: 1.0: This field indicates the log file format.
The remaining fields are listed and described in Table 1.1 lists of the example

Field	**Appears As**	**Description**
date	2002-05-24	This log file entry was recorded on May 24, 2002.
time	20:18:01	This log file entry was recorded at 8:18 P.M. UTC.
c-ip	172.224.24.114	The IP address of the client.
cs-username	-	The user was anonymous.
s-ip	206.73.118.24	The IP address of the server.
s-port	80	The server port.
cs-method	GET	The user issued a **GET**, or download, command.
cs-uri-stem	/Default.htm	The user wanted to download the contents of Default.htm.
cs-uri-query	-	The URI query did not occur.
sc-status	200	The request was fulfilled without error.
sc-bytes	7930	The number of bytes that the server sent to the client.
cs-bytes	248	The number of bytes that the client sent to the server.
time-taken	31	The action was completed in 31 milliseconds.
cs(User-Agent)	Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+2000+Server)	The type of browser that the client used, as represented by the browser.
cs(Referrer)	http://62.224.24.114/	The Web page that provided the link to the Web site.

Table 1.1 describes the web log fields

Since not all browser requests succeed ,the status code field provides a three-digit response,from the web server to the client's browser, indicating the status of the request,a sample of the possible status codes that a web server could send follows.
1-Successful transmission (200 series): Indicates that the request from the client was received, understood, and completed.

- 200: success
- 201: created
- 202: accepted
- 204: no content

2- Redirection (300 series): Indicates that further action is required to complete the client's request.

- 301: moved permanently
- 302: moved temporarily
- 303: not modified
- 304: use cached document

3-Client error (400 series): Indicates that the client's request cannot be fulfilled, due to incorrect syntax or a missing file.

- 400: bad request
- 401: unauthorized
- 403: forbidden
- 404: not found

4- Server error (500 series): Indicates that the web server failed to fulfill what was apparently a valid request.

- 500: internal server error
- 501: not implemented
- 502: bad gateway
- 503: service unavailable

After the previous explanation of web log file, that ensures web log file can help answering question about user behavior like what are pages are most and least popular, which browsers and operating systems are commonly used which is the most traffic time.

Marketing companies and E-commerce web sites look forward to build predictive systems and recommendation engines based on web log file data , whenever the system is more accurate to discover patterns and predict values , whenever that demands data mining techniques implementation such as association rules ,sequential patterns ,clustering and classification, in addition to find the most interesting rules or the important results requires analysis tools such as visualization techniques data and knowledge query.

1.3 Scope of research

In this thesis the proposed system has a number of constraints, the first one is that the web log file is simulated and similar to web log file in recommendation engines in E-commerce web sites ,the second one the web log file records are more 1000 records, the third constraint the proposed may use two data mining techniques (descriptive & predictive methods),artificial intelligence technique and BigData analytics are not included, the last constraint the output of data mining techniques are analyzed and visualize in 2-D charts by Excel.

1.4 Thesis Objectives

The objectives of this thesis can be concluded in the following:

1- Prepare the web log files for preprocessing operations.

2- Analyze web log files using data mining techniques.

3- Discover correlated web pages, groups of users with similar interests and behaviors using web mining techniques and algorithms.

4- Identifying the typical user behavior navigation

1.5 Software tools used in research

In the case study we will illustrate the programming language and tools which achieved our system these programming language and tools are briefly as follows:

- ***C#:*** C# is an interpreter; high-level programming language was used practically in every part of the system.
- ***Weka:*** is a data mining and a machine learning tool for data pre-processing, classification, regression, clustering, association rules, and visualization.
- ***Software Tools and Frameworks:*** we will use a variety of readily-available software tools and frameworks to deal with the incidental tasks of software development and be able to concentrate on the main objectives of this research. These tools are listed as follows:

    - LINQ: **Language Integrated Query** is a Microsoft .NET Framework component that adds native data querying capabilities to .NET languages.
    - Log Server file history files from variety of sever logs platform.
    - **Notepad** is a simple text editor for Microsoft Windows. typically saved with the .txt extension
    - Microsoft Excel s a spreadsheet program which allows us to create log flat files holding the entire web logs files collecting for different platforms.

On the other hand, we created our database by downloading log files dataset in notepad format from the targeted website .Then we converted log files dataset into the required extension of (csv,xls,xlsx). Next step, we used excel to manipulate data into the right order and format, after that errors, missing values and the irrelevant data was removed by using the Weka software solution to make sure of the data capability to be readable. LINQ was used to query the excel file database to select the proper data. In this case the dataset was ready to apply an association rule algorithm called Class *aprori algorithm* to a group of pages that are accessed together. We integrate the classed rules with classification to classify the users who accessed web pages to one of predefined classes to represents a number of similar cases or the number of items in a single group. Both of the previous algorithms were programmed using C# programming language.

1.6 Thesis Organization

The thesis is organized into the following chapters:

Chapter 1: is the introduction of the thesis which introduces the definition of web mining. The definition of web usage mining, web usage mining ,stages ,pre-processing stage, pattern discovery, pattern analysis stage and their methods to perform web usage mining process .

Chapters 2: background of the consumer and user behavior concepts model and their relationship with data mining in this chapter we will provide a brief introduction about the model of consumer behavior , the techniques of data mining to improve the quality of the model ,the data being used and the benefits of using data mining techniques.

Chapter 3: conducts the methodology used, which explains how the phases of web usage mining used such as association rules method and the algorithm called *class aprori algorithm* as well as classification algorithms applied in thesis work.

Chapter 4 : describes the implementation including the association rules and classification algorithms used in the thesis work, as well as the tools (software's) which evaluate the demonstrated experiments.

Chapter 5 : this chapter will summarized the results and conclusions of the thesis.

Chapter 2
Background

Chapter 2

2.1 Introduction

User behavior of the web is considered as possible customer or consumer in the market for the company's products that what brings the need to understand the user behavior to contribute in a process which is building better model of consumer behavior.

In return building a model of consumer behavior will assist in developing and distributing product as well as getting the right price point and improving successful promotional activities [8].

The profile of mental procurement process has been studied extensively to gain knowledge that could be useful for companies to be more successful, the company's missions to understand the procurement process, to match the marketing activities that will help the company's customers get the kind of contact at the right time. To get a clear vision of how consumers behave in buying mode [8]. In this chapter we will explore briefly the model of online consumer behavior.

2.2 Online Consumer Behavior

Significant consumer behavior is not simple, consumers may say one thing but do another, and they may be exposed to influences that change their mind at the last minute. Many researchers describe consumers behavior as the study of individuals or groups and the mental emotional and physical process the use to select, obtain, consume and dispose of products or services, to satisfy needs and desires and the impact that theses processes have on consumer society [9,10].

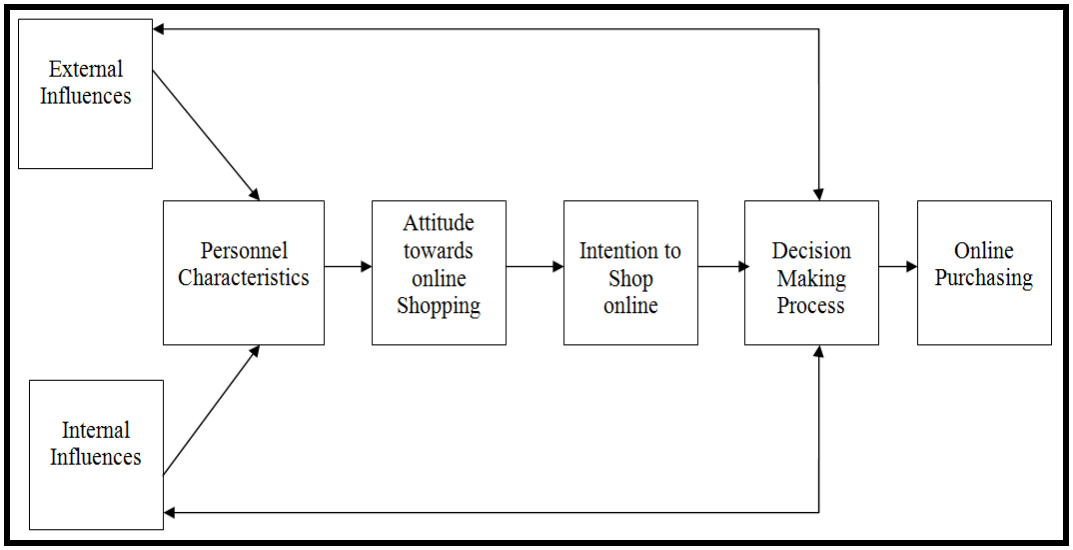


Figure 2.1: the model of online consumer behavior

This model is broken into six factors listed as follows:

1- Personnel characteristics.

2- External influences.
3- Internal influences.
4- Attitude towards online shopping.
5- Intention to shop online.
6- Decision making process.

An explanation of each factor is briefly conducted in the following sections.

1. Personnel Characteristics: A Personnel characteristic refers to how people live? , how they spend their time and money? , what activities they pursue and their attitudes and opinions about the world they lives in? Personnel characteristics is affected by both internal and external influences, in exchange Personnel characteristics affects the attitude towards online shopping [9,11].

2. External Influences Factors: the external influences will be divided into the following sub factors:

A. Culture: Culture is defined as the complex whole that includes knowledge, belief, art, law and any habits; finally it influences the person thoughts feeling and behavior [9].

B. Demographic and Geographic Factors: Demographics is the study of characteristics of a human population, demographic categories include age, gender, race, education and income. Geographic is the study of characteristics is the study of characteristics of geographical region, geographies is consisted of the following elements, population size, density and region [9].

C. Reference Groups Factors: References groups are generally defined as groups are being used by individuals as the basis for his or her current behavior, beliefs and feelings. References groups play a huge role in the lives of young customers as they are easily modeled into consumers depending on their references groups, reference group are generally classified into four categories:

1- Primary groups: family and friends are considered as most influential.
2- Secondary groups: they have limited face to face interaction and are less comprehensive and influential community organization and schools.
3- Inspirational groups :exhibit a desire to adopt the norms ,values and behavior of others with whom the individuals aspires to associate, such as media advertisement ,models, sport athletics ,media stars.
4- Dissocialize groups: these groups are less desirable appeal individuals can be seen to reject their values and behavior [8].

D. Consumer Resources: Consumer resources represent all the physical , psychological and materials methods which consumer has to draw on including income ,self confidence ,health ,intelligent ,energy level ,eagerness to purchase these different forms have been analyzed under three categories:

1- Economic resources: economic resources represent financial means available to consumer.
2- Cognitive resources :refers to the mental capacity available for undertaking various information processing activities [8,9]

3. Internal Influences Factors: the internal factors indicate the following sub factors:

A. Perception, Memory and Learning: These three combined factors play role in how consumer perceive and remember information activate process which begins at young age and is essential for a child to be successfully socialized ,e.g. as a child develops he/she starts to orientate themselves consistently towards concepts and objects[11].

B. Motivation: Motivation is considered as creation representing an unobservable internal force that stimulates and compels a behavioral response and provides precise direct to that response. A person is said to be motivated when his/her system is aroused and driven towards a behavior in satisfying a desired goal [9].

4. Attitude Toward Online Shopping: Attitudes refers to consistent favorable and unfavorable towards objects [10]. Attitudes consist of three components listed as follows:

1- The cognitive component holds person knowledge and beliefs about a product.
2- The affective component represents a person's feeling about a product.
3- The behavioral component refers to a person's action is meant to a product.

It is believed that consumers will affect intention to shop online and eventually whether a transaction is made [8]:

First: it refers to the consumer acceptance of the internet as shopping channel.

Second: it refers to consumer attitudes toward specific internet store; i.e to what is appealing of shopping at this store? [11].

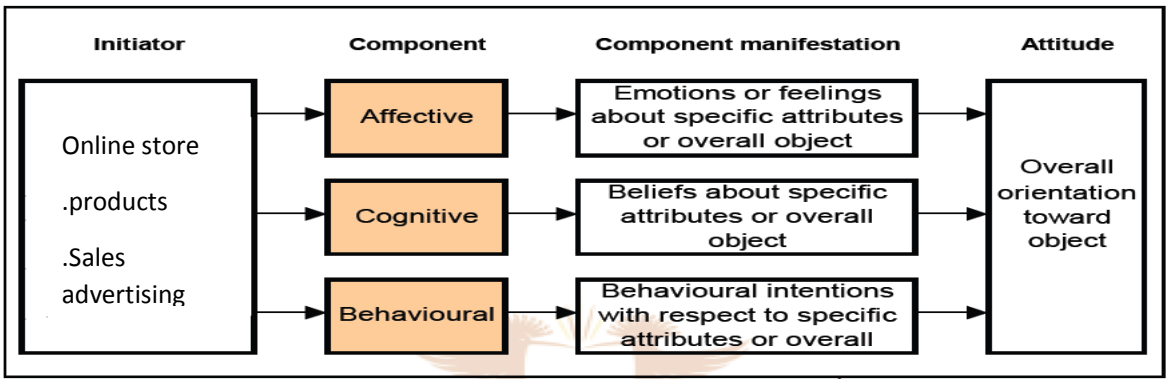
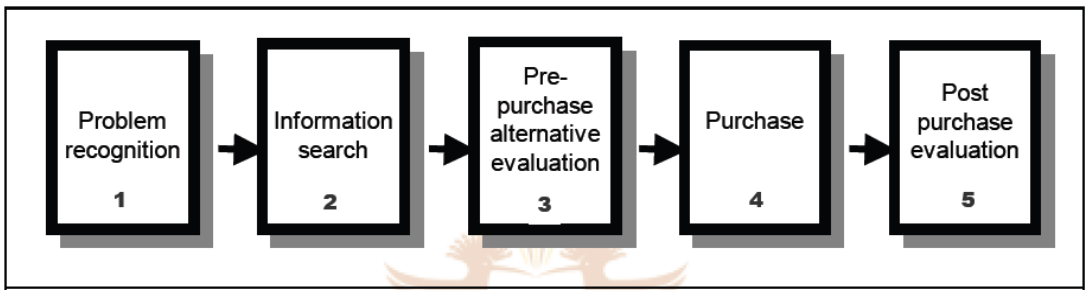


Figure 2.2: Attitudes components and manifestations toward online shopping

5. Intention to Shop Online: This process is measured by consumer's willingness to buy and return for another purchase [12]. As indicated in figure 2.1, consumers, intention to shop online is positively associated with attitude towards Internet buying, and influences their decision-making and purchasing behavior.

6. The Decision Making Process: The decision making process has the ability to help explaining the manner in which consumers act in the market place. Figure 2.3 illustrates the decision making process in steps.



Figurer 2.3: steps of decision making process

A. Problem Recognition: Problem recognition or need recognition is the first step of the consumer decision making process ,when individuals realizes a difference between what he or she perceives to be the ideal or desired state (the situation the consumer wants to be in) of affairs compared with the actual state (the consumer's current position) of any point in time . In other words, it is state of desire that initiates a decision process that in turn occurs through the interaction of individual difference and environmental influences [9].

B. Search for Problem Solving Information: This section is combined of two issues search and information search that can be defined as the motivated activation of knowledge stored in memory. Information is considered as the knowledge as the consumer currently possesses. In order for a consumer to make a meaningful decision when making a purchase, he/she will need some direction to what type of information is required, in helping consumers during their information search, numerous sources of information are expressed as follows:

- The internet.
- Personnel sources.
- Experimental sources.
- General purpose media [9,13].

C. Pre-Purchase (Alternative Evaluation): It is a process which the alternative choices are evaluated and selected to meet the consumer needs [8].

D. Purchase: Once an alternative is chosen and final decision has been made the consumer then moves to purchase step ,the consumer then attempts to put his thoughts into action the consumer must address in

executing a purchasing action ,such as whether to buy ?,when to buy? , what to buy? , how to pay? [9,14].

E. Post Purchase Evaluation: This step deals with the way that consumer evaluates uses or consumes products after purchasing it, the consumer usually examines the consequences of his/her purchase the result may be satisfying or unsatisfying. Unsatisfying customer is often resulting of prevailing cognitive dissonance [9,15].

F. Online Purchasing: refers to consumers, actions of placing orders and paying. This is the most substantial step in online shopping activities, and Internet store sales examination of the relationship between online purchasing behavior, perceived ease of use, perceived usefulness, perceived risk of the product/service, and perceived risk in the context of the transaction. Online purchasing is reported to be strongly associated with the factors of personal characteristics, attitudes toward online shopping, intention to shop online, and decision making [9,12].

2.3 Web Mining Impact on Consumer Behavior Online

Marketing does not stop at understanding the buying processes of the customer. However, company need to understand their buying patterns and the market in which they operate. In the next a few sections we will mention to the most recent techniques and methods to improve the buying process of consumers by associating it with understanding user behavior applying web mining techniques [10].

1. User Behavior and Data Mining

Most marketers understand the value of collecting customer data, but also realize the challenges to take advantage of this knowledge to create intelligent, proactive pathways back to the customer. Techniques of data mining techniques for recognizing and tracking patterns within data which helps businesses sift through layers of data unrelated to what appears to meaningful relationships, where they can anticipate, rather than simply react to customer needs, and also extract data could impose redefine relationships with customers [8].

2. Web Mining and Customer Relationships

Web mining is one kind of these techniques that efficiently handle the tasks of searching the needed information from the internet, improving the web site structure to provide better internet service quality and discovering the informative knowledge from the internet for advanced web applications. Web mining could be categorized into three types such as web content, web structure and web usage mining. In this study, we focus on web usage mining in the since of discovering user access pattern knowledge from web log files, which contain the historic visiting records of users on the website [16].

Moreover, web usage mining considers the way which companies interact with their customers change dramatically over the past few years. No longer

guarantee the continuation of the work of the client. As a result, companies have found that they need to better understand their customers, and respond quickly to their needs and desires. In addition, the time frame you need these responses to be shrinking. It is no longer possible to wait until the signs of customer dissatisfaction clear before action must be taken. To succeed, companies must be proactive and anticipate what the customer desires [8].

2.4 Datasets Applied in Web Mining Application

Click Stream Data: is the path that the user creates when steering through the sites and following links. It can be used to evaluate the traffic and popularity of the page.

Shopping Chart: can provide information in e-business where the purchases were made and where the customer left the order unfinished.

Psychographic Data: would include data on user's attitudes towards topics, products etc., buying behavior and beliefs.
Access Data: it counts the time between the last and next access to the same URL.
Time Data: gives information on amount of time a user spends exploring the site, the product or topic he or she is interested in [8].

2.5 The relationship between web mining and society:

Web mining has a distinct role in responding to the daily challenges in service sectors such as health, education and entertainment which leads to the main beneficiaries of social services sectors are affected of web mining are reviewed as follows:

- **E-Learning** : The learning process undergone significant radical changes over the years by means of progress in the web mining which evolved the traditional learning to E-Learning in return facilitated the missions of researchers and students access to the owners of experience in the areas of knowledge fields through research forums on the World Wide Web and professionals networks groups.
- **E-Government** : web mining is used by E-government to take decisions , making it easier for the government more effectively in related to its policy for maintaining transparency at the national and international level.
- **Digital Libraries:** Since starting educational institutions to convert traditional approaches such as books research papers , magazines and newspapers to a digital format, all this requires a means to facilitate access to these digital documents represented in web mining, mutually digital libraries have played a role in organizing, structuring and indexing information within the Web that contains large amounts of semi-structured documents (not organized in tables) contributing to web evolving.
- **Public security and crime investigation**: web mining plays a role in the fight against illegal activities by dividing the wide web into two parts.
  1- The white list : includes web sites containing secured information on the web.
  2- The black list : including piracy within the web, virus spreading ,online gambling, hacking, E-commerce faulty web sites[5].

2.6 Benefits of using web mining in systems for marketing strategies

Usage mining allows companies to produce productive information pertaining to the future of their business function ability, some of this information can be derived from the collective information of lifetime user value, product cross marketing strategies and promotional campaign effectiveness. The usage data that is gathered provides the companies with the ability to produce results more effective to their businesses and increasing of sales. Usage data can also be useful for developing marketing ills that will out-sell the competitors and promote the company's services or product on a higher level [17],one of the systems that organize the process between the consumers and the companies is customer relationship management (CRM) which stand of establishing a system manages relationships with customers, with the expansion of companies in terms of number of branches, and increasing divisions Administrative, bringing There is an urgent need for this system to reduce the cost of communications, and the use of automation to work, so The system is the one who coordinates the things the customer without the need to develop a specialized staff to follow up customers, and this necessity ,dictated by increased administrative divisions within the company itself CRM solutions use intelligent systems to analyze data, identify the demographic profiles, measure the buying capacity and other unknown behavioral patterns of data about the customer and based on all that take decisions on behalf of organizations[19].

Also online shops need to offer personalized products to clients but before being able to do that they have to personalize the web sites to the clients. This is where the web mining techniques in web server logs are coming in. Companies can use the basic data retrieved from the data logs to analyze customer behaviors, evaluate the current usage, if the customers liked or disliked it and so on. Creating an adaptable web site to each user, first, the user navigation patterns in the web have to be found and analyzed. Web mining is a method extracting valuable information from the data for statistical purpose, utilizing web mining methods to induce and extract useful information from web information, services and goods online increases, web mining activities that can expand rapidly allowing firms to retrieve highly personalized data about customers [8],for instance recommendation engines are the closest example of taking the advantage of user data, the recommender systems, suggest content based on previous behavior or purchases. Such systems typically use a predictive model based on a user's previous interactions such as products purchased or item characteristics and suggests content with similar elements. Amazon, Netflix and music services, recommendation engines attempt to discover and apply patterns in data by learning consumers preferences and adapting brand experiences to their needs or interests, the site Netflix.com is a product recommendation site which works on the web mining" concept, the site gives recommendations regarding the various movies based on their rental and user profiles, recommendations based on earlier movies recommended by user[20], the output of web mining techniques like Association, Classification, Clustering produce some patterns that may be the input to the Recommendation systems Engine which is one of the application areas of the Web usage and gives the ability to predict the next visited page or next product for a given user.

At last the web mining keeps on growing to produce more efficient applications and services are capable of handling data with much bigger size as BigData hadoop,Machine learning windows azure and No SQL services which can push the companies to cut costs and attract more customers which leads to more benefits include more accurate results, improved business decisions, improved marketing strategy ,revenue increased due to increased customer base and lower production costs.

Chapter 3
Methodologies

Chapter 3

3.1 The Proposed Methodology Used

In order to understand the user behavior, we use web server log files and meta data describing page contents to extract sessions identification of the web user in the preprocessing phase. data mining techniques are integrated such as association rules and classification to build a classifier assists in forming a user profile.

In particular a user profile describes a set of user interests which can be molded via categories for instance, like sports, technology or education to get useful results which made the surfing in the web more beneficial [21].

The proposed system used the classification based on association algorithm (CBA) which is consisted of two phases. First phase the class *Aprior* rules based on *Aprior* algorithm are generated for finding association rules called (CBA-CARs),the second phase is the classifier builder called (CBA-CB) which aims to build the classifier by using the generated CARs of the previous phase. The classification result is the most access pages in their categories.

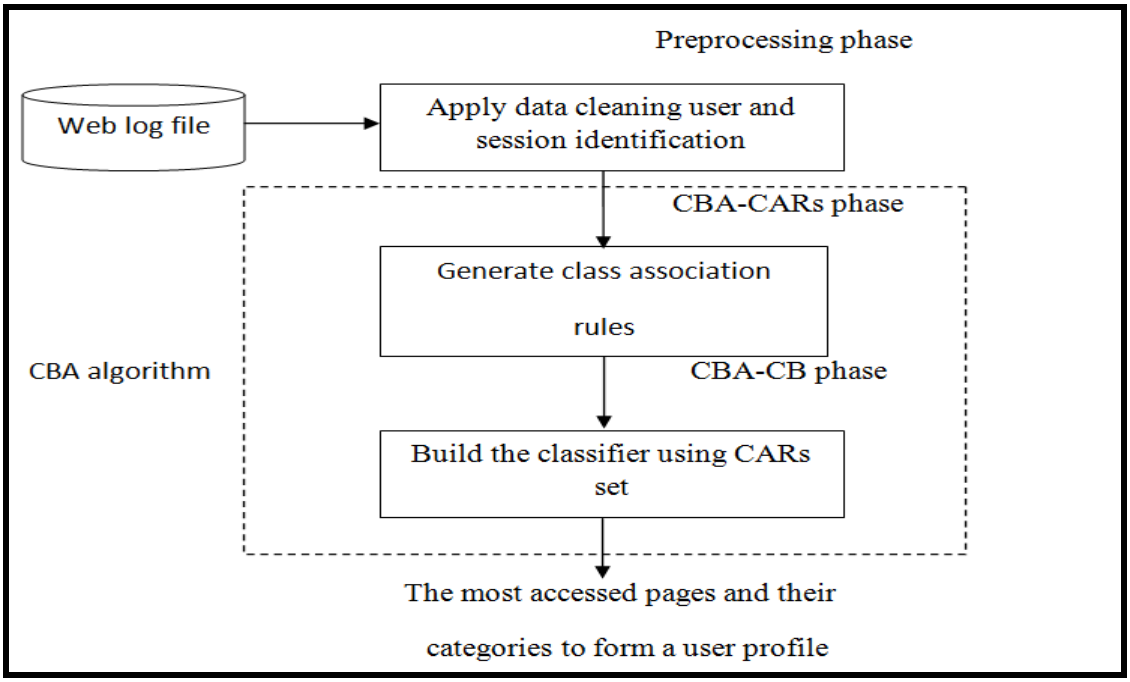


Figure 3.1: the Phases of the proposed system

The proposed system consists of the following three phases as mentioned in the

figure 3.1 in details as following:

3.2 Preprocessing Phase:

Usually every single user has a wide information interests to his private or professional interests to his private or professional life which can be modeled by categories (class type). In this phase we will rely on a number of different information whether different types of information, or sessions, or attributes of log-life. But in some cases, some of them are mandatory and some are optional based on some circumstances [21].

Every user request generates an entry in the servers log files indicating his interest in the specific page and its contents these log files entries preprocessed to be ready to next phase to derive user profile characteristics[22].

Data is gathered from individual web sites, for instances, sport, news ,social ,technology ,education and business ,where every web page site has a name which could be found at the HTTP header e.g. via meta-tags <meta name =”description” content=”football”> which belonged to sport category (class type) or web site. On the other hand, usually a user clicks his way through the pages he interested and web servers generates corresponding log data which can be used to model his behavior, i.e. “*we need access to all relevant web server log files”* .

An alternative way to get a comprehensive view on user interests is shown in figure 3.2b. In this scenario all web traffic is handled by a web proxy extracting user sessions identification form proxy log file which gives more complete view on user behavior [23].

This brought an advantage and disadvantage of preprocessing the log files in web usage mining benefits e.g. generating recommendation system, personalizing web site evaluation. But it also has disadvantages of log files preprocessed that have to be cleaned which they are possibly worthless for the purpose of user profiling. Since cleaning data is the most essential step in the preprocessing phase [21].

1. **Data Cleaning**

First of all, irrelevant data should be removed to reduce the search space and to skew the result space. Since the intention is to identify user sessions, build up out of page views, not all hits in a log file are necessary, this is true for server logs as well as for proxy logs, a log file generates a hit for every requested file. Since a HTML-page may consist of several files (text, pictures, sounds, and several frames) it would be useful if we could keep only a single hit for each page view [24].

To get an idea of user behavior, it is only necessary to keep track of the files that the user specifically requested. Very often, all hits with a suffix like .jpg, .gif, .wav, etc. are removed out of the log file and what is left will be placed in the cleaning step after

removing the data not important to prepare log file for further processing. After all we extract the following records from the targeted log file [25].

- **User_ID:** It represents the IP address whish is used to extract user and session information.

- **Requested Pages**: The reveals of what information was accessed, we extract success full GET and POST request only while failed requests are not used because they are not relevant for user profile and are eliminated, thus we remove all log-file entries with HTTP statue code other than 200 (successful request).

- **Page content(meta-data tags):** It illustrates the page meta data that can facilitate knowing web page content and the class of web page which could be extracted from one of the following HTML tags [22].the page contents meta information structure are as follows:

  1- From page <title>page<title>.
  2- Description tag the second most important tag after the title Tag because it has strong relations between hyperlink page, title page,

  meta description and page content <meta name="description",

  content="Entertainment">.

  3- Key words tag which defines key words for web page which is provided in HTML header <meta name="key words" content="Music">.

  All Meta information exists in HTML head section of web page source.

- **Date_Time:** The time of accessing to the web page we use it to determine the period of session [20,22,].

2. User Identification

User identification step starts when the log files are cleaned. This step is the next step in the data preprocessing. User identification step could be summarized in the following two steps:

1- Converting the IP address to a domain name.

2- The web server randomly assigns an ID to web browser while it connects first time to the site. This is called cookies. The web browser sends the same ID back to web server effectively telling the web site that a specific user has returned.

Cookies help the website developer to easily identifying individual visitors which results in a greater understanding of how the site is used [26].

In this work, we will assume that each combination of IP address / Agent / Operating system represents a single user. We will add the login information to the log file to get a username.

3. Session Identification

A session is a set of page references during one logical period. Historically a session would be identified by a user logging into a computer, performing work and then logging off. The login and logoff represent the logical start and end of the session. The activities of a single user from the web log files is called a session [5].

Session identification aims to split the page access of each user into separated sessions. It defines the number of times the user has accessed a web page and time out defines a time limit for the access of particular web page for more than 30 minutes if more the session will be divided in more than one session.

User:pc1

Session1

05:10	http://www.databaseanswers.org	pc1	Education
05:22	http://database.firstnormalform.htm	pc1	Education

Session2

07:43	http://www.aljazeera.net/news/arabic	Pc1	News
07:44	http://www.aljazeera.net/news	Pc1	News
07:50	http://www.complete-review.com/	Pc1	Social

User2:pc2

Session1

05:13	http://www.dotnetperls.com	Pc2	Education
05:17	http://www.dotnetperls.com/data	Pc2	Education
05:26	http://www.java2s.com	Pc2	Education
05:30	http://www.java2s.com/.htm	Pc2	Education

Session2

07:30	http://www.ferryhalim.com/orisinal/	Pc2	Entertainment
07:33	http://www.ferryhalim.com/oris.htm	pc2	Entertainment

Figure 3.2: Example of session identification

The session identification procedure is summarized as follows:

1- For each distinct user identified in the preceding section, assign a unique session ID.

2- Define the timeout threshold $t < 30\ min$.

3- For each user, perform the following:

- Find the time difference between every two consecutive web log entries.
- If this difference exceeds the threshold *t*, assign a new session ID to the later entry.

4- Sort the entries by session ID [26].

3.3 The Classification Based on Association Rules – Class association rules (CBA-CARs) phase

In this phase classification and association rule discovery are integrated in the proposed system, since data mining plays an important techniques in discovering user behavior.

A combination of association rule mining and classification are applied in the proposed system since both techniques concerned with finding rules that accurately predict a single target (class) variable. the key strength of association rule mining is that all interesting rules are found [27].

Applying the association rule into classification can improve the accuracy and obtain some valuable rules and information that cannot be captured by other classification approaches. Both classification rule mining and association rule mining are indispensable to practical applications. The integration is done by focusing on a special subset of association rules whose right-hand-side is restricted to the classification class attribute. Our objectives are to generate the complete set of CARs that satisfy the user-specified minimum support (*minsup*) and minimum confidence (*minconf*) constraints and to build a classifier from the CARs[28]. the previous techniques are explored in details as follows.

3.3.1 The Association Rule Mining

Association rule mining is an important technique to discover hidden relationships among items in the transaction, finding frequent patterns, associations, correlations of set items of objects in web log transaction databases in the term of the web usage mining [29]. the association rules refer to a sets of pages that are accessed together with a support value exceeding some specified threshold. The support is the percentage of the transactions that contain a given pattern. the web designers can restructure their web sites efficiently with the help of the presence or absence of the association rules. When loading a page from a remote site, association rules can be used as a trigger for perfecting documents to reduce user perceived [6].

Basic Concepts of Association Rules

$I = \{i_1, i_2, i_3, \ldots\}$, Where I is a set of items.

$T = (t_1, t_2, t_3, \ldots)$, T is a set of transaction

$$t_i \subseteq I$$

Let's suppose

$X \rightarrow Y$, where $X \subset I$, $Y \subset I$, $X \bigcap Y = \emptyset$

X or Y is a set of items called itemset

e.g. a transaction may be $t_i = \{beef, chicken, cheese\}$

$$beef, chicken \rightarrow cheese$$

Where i.e $X \rightarrow Y$

A transaction $t_i \in T$ it contains *X* if *X* is a subset of t_i the support count of *X* in *T*(denoted *X.count*) is the number transactions in *T* that contains *X*.

The strength of a rule is measured by its support and its confidence.

Support: the support of a rule ,$X \rightarrow Y$,is the percentage of transactions in *T* that contains $X \cup Y$ .The rule support determines how frequent the rule is applicable in the transaction set *T*. Let *n* be the number of transactions in *T*.

The support of the rule $X \rightarrow Y$ is computed as follows

$$support = \frac{(X \cup Y).count}{n} * 100\%$$

Confidence: the confidence of a rule, $X \rightarrow Y$ is the percentage of transactions in *T* that contains X also contains *Y* .

It is computed as follows:

$$confidence = \frac{(X \cup Y).count}{X.count} * 100\%$$

Association rule mining is a two steps process:

1. **Find all frequent itemsets**: By definition, each of these itemsets will occur at least as frequently as a pre-defined minimum support count.
2. **Generate strong Association rules from the frequent itemsets:** By definition, these rules must satisfy minimum confidence [30].

3.3.2 The Aprior Algorithm

Apriori is an influential algorithm for mining frequent itemsets for boolean association rules. The name of the algorithm is based on the fact that the algorithm uses prior knowledge of frequent itemset properties.

Apriori is a common association rule algorithm which orders rules according to their confidence and uses support as a tiebreak. Given a set of item sets (for instance, sets of website transactions, each listing individual pages visited), the algorithm attempts to find subsets which are common using minimum confidence of the item sets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time, and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found [6],figure 3.3 shows the pseudo code in the following page.

$F1 = \{$frequent 1-itemsets$\}$;
for $(k = 2; Fk-1 \neq \Phi; k++)$ **do begin**
$Ck =$ apriori-gen$(Fk-1)$; //New candidates
foreach transaction $t \in D$ **do begin**
$Ct =$ subset(Ck, t); //Candidates contained in t
foreach candidate $c \in Ct$ **do**
$c.count++$;
end
$Fk = \{c \in Ck \mid c.count \geq minsup\}$;
end
$F = \cup k\, Fk$;

Figure 3.3: Apriori Algorithm pseudo code

3.3.3 Aprior example

This example is based on table 3.1 below called AllElectronics transaction database D.

TID	**TID List of item IDs**
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3

T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1,I2,I3

Table 3.1 representsTransactional Data for an AllElectronics

there nine transaction in this database, |D|=9 as shown in figure 3.4

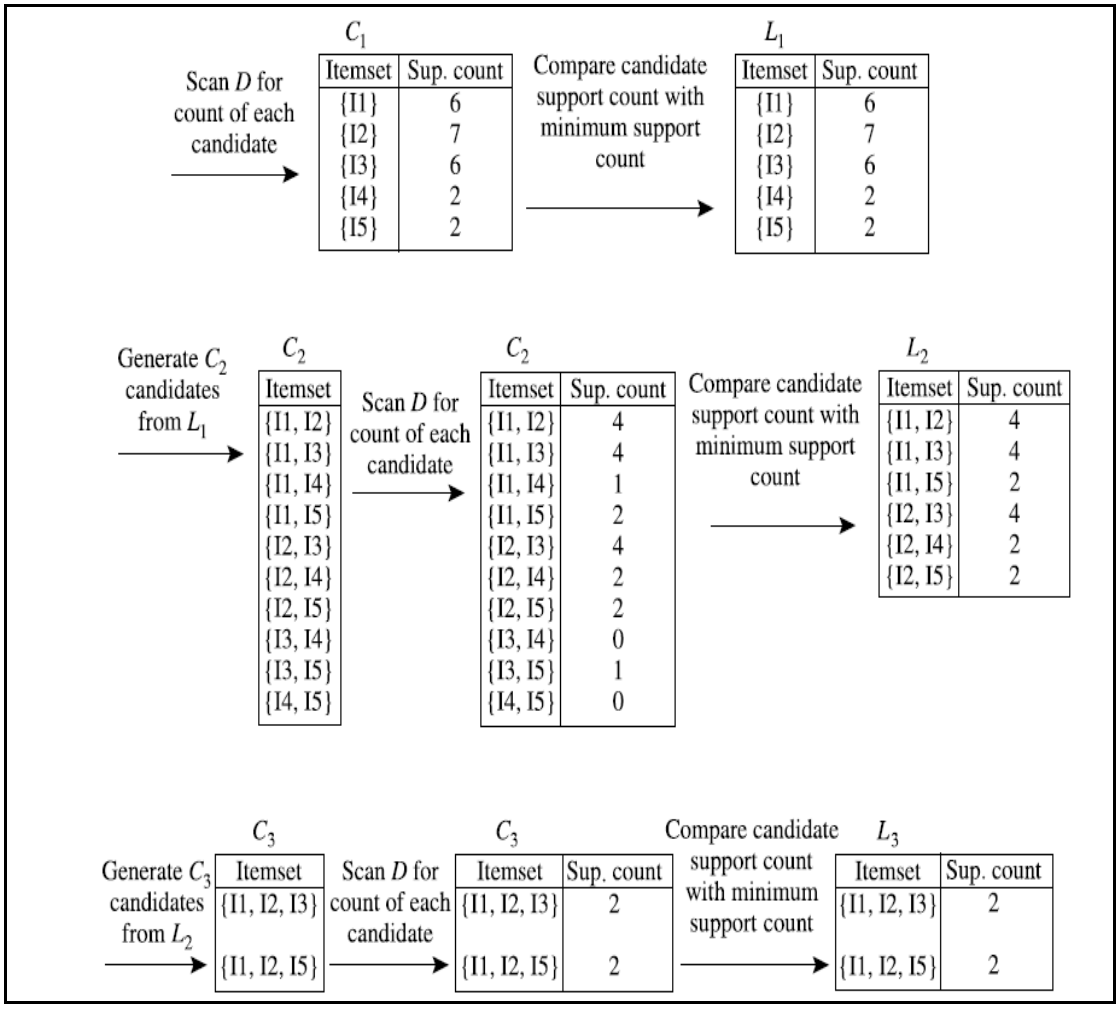


Fig 3.4 Generation of the candidate itemsets and frequent itemsets, where the minimum support count is 2.

In addition minimum support count is used to remove that is used to remove the least frequent itemset from the generated itemset as explained in figure 3.4 in the following steps.

1- Distinguish 1-itmeset at table 3.1 {I1,I2,I3,I4,I5} then count sup.count of each 1-itmeset ,the1-itemset is less than 2 sup.count is discarded .

2- From the frequent item set at the previous stage generate 2-itemset which are

{I1,I2},{I1,I3},{I1,I4},{I1,I5},{I2,I3},{I2,I4},{I2,I5},{I3,I4},{I3,I5},

{I4,I5}},after scanning D for counting of each 2-itemsets candidate have sup.count less than 2 as shown in figure 3.4 at this case {{I1,I4},{I3,I4},{I3,I5},{I4,I5}}

3- At the case of generation 3-itemset from the frequent 2-itemset which are {{I1, I2, I3}, {I1, I2, I5}, {I1, I3, I5}, {I2, I3, I4}, {I2, I3, I5}, {I2, I4, I5}},then the aprior prune property is used which is about all nonempty subsets of a frequent itemset must also be frequent,in other words all the generated itemsets that contains non frequent subsets must be removed e.g at case generated contained 3-itemset {I1,I2,I3} the subsets 2-itemsets are {I1,I2},{I1,I3},{I2,I3} all the previous subsets are members of frequent 2-itemset ,therefore {I1,I2,I3} is considered member of candidate generated 3-itemset.

At the case 2-itemsets subsets of {I1,I3,I5} are {I1,I3},{I1,I5} and {I3,I5},where {I3,I5}is not frequent itemset ,in return {I1,I3,I5} is considered a member of candidate generated 3-itemset so it is removed after checking all possible 3-itemset using prune property, the result of pruning 3-itemset {{I1,I2,I3},{I1,I2,I5}} the generated 3-itemset are compared with D to check occurrence or 3-itemset in D to get sup.count of each as shown in figure 3.4 the result of frequent 3-itemset is {{I1,I2,I3},{I1,I2,I5}}.

4- Generating association rules can be achieved once the frequent itemsets from transaction in a database D,It is straight forward to generate strong association rules from them this can be done using formula .

Confidence(A=>B)=P(B/A)=support_count(AUB)/ support_count(A)

Where support_count(AUB) is the number of transaction containing the itemsets AUB and sup_count(A) is the number of transaction containing the itemset A.

Based on the previous relationship association rules can be generated as follows:

a-For each frequent itemset f, generate all nonempty subsets of f.

b- For every nonempty subset s of f, output the rule "s=>(f-s)"

if support_count(f)/support_count(s) ≥ min conf,

where min_conf is the minimum confidence threshold.

e.g The data contain frequent itemset x= {I1, I2,I5},the subsets of x are .What are the association rules that can be generated from X? The nonempty subsets

of X are {{I1, I2}, {I1, I5}, {I2, I5}, {I1}, {I2}, and {I5}.

The resulting association rules are as shown below, each listed with its confidence:

{I1, I2}=>I5, confidence = 2/4 = 50%

{I1, I5}=>I2, confidence = 2/2 = 100%

{I2, I5}=>I1, confidence = 2/2 = 100%

{I1}=>{I2, I5}, confidence = 2/6 = 33%

{I2}=>{I1, I5}, confidence = 2/7 =29%

{I5}=>{I1, I2}, confidence 2/2 = 100%

If the minimum confidence threshold is 70%, then only the second, third, and last rules are output, because these are the only ones generated that are strong.[3]

3.3.4 Class Association Rules Mining

The use of association rules focus is to produce association rules that have only a particular attribute in the consequent (right hand side of the rule). in the prosed system interested in only rules with fixed target items in the right hand side, because that assists to understand the user behavior of the web. These association rules produced are called class association rules (CARs), which are useful because many types of the web data are in the form of transaction e.g. search queries and pages clicked by users.

Such system often have target items, e.g. advertisements, web sites want to understand user activities are related to advertisements that the user may click or view[31,32] ,this indicates to the issue of classification or predication which is about to be mentioned in details at the next sections explaining how to integrate class association rules with classification.

Basic Concepts of Class Association Rules

Let *D* be the transaction dataset of *n* transaction.

Each transaction is labeled with a class *y*

Let *I* be the set of all items in *D*, and *Y* be the set of class labels (or target items).

Where $I \cap Y = \phi$

We say that a data case $i \in D$ contains $X \subseteq I$, a subset of items, *if* $X \subseteq I$.

A class association rule (CAR) is an implication of the form: $X \rightarrow y$

Where $X \subseteq I$, and $y \in Y$.

A rule $X \rightarrow y$ holds in *D* with confidence *c* if *c*% of cases in *D* that contains *X* are labeled with class *y*.

The rule $X \rightarrow y$ has support *s* in *D* if *s*% of the cases in *D* contain *X* and are labeled with class *y*.

In general, a class association rules is a different form of normal association rules in two cases:

1- The consequent (right hand side) of a CAR has only a single item, while the consequent of a normal association rule can have any number of items.
2- The consequent y of CAR can only be form the class label set *Y*, i.e. $y \in Y$, no item from *I* can appear as consequent ,and no class label can appear as a rule condition.

In contrast, a normal association rule can have any item as condition or consequent.
Our objectives are as follows:

I. To generate the complete set of CARs that satisfy the user-specified minimum support (called *minsup*) and minimum confidence (called *minconf*) constraints.
II. To build a classifier from the CARs, The key is to find all *ruleitems* that have support above *minsup* [28].

3.3.5 The Class Aprior Algorithm

The key operation is to find all *ruleitems* that have support above *minsup*.

A *ruleitem* is of the form: (*condset*, *y*)

Where, *condset* is a set of *items* $condest \subseteq I$ and $y \in Y$ is class Label

The support count of the *condset* (called *condsupCount*) is the number of cases in *D* that contain the *condset*.

The support count of the *ruleitem* (called *rulesupCount*) is the number of cases in *D* that contain the *condset* and are labeled with class *y*.

Each *ruleitem* basically represents a rule:

$$condset \rightarrow Y$$

whose *support* is :

$$support = \frac{rulesupCount}{n} * 100\%$$

Where *n* is total number of transactions in *D*

And whose *confidence* is:

$$confidence = \frac{rulesupCount}{condsupCount} * 100\%$$

The class Aprior algorithm generates all the frequent *ruleitems* by making multiple passes over the data, in the first pass; it counts the support of individual *ruleitem* and determines whether it is frequent.

In each subsequent pass, it starts with the seed set of *ruleitems* found to be frequent in the previous pass. It uses this seed set to generate new possibly frequent *ruleitems*, called *candidate ruleitems*.

The actual supports for these candidate *ruleitems* are calculated during the pass over the data, at the end of the pass, it determines which of the candidate *ruleitems* are actually frequent. From this set of frequent *ruleitems*, it produces the rules (CARs).

Let *k-ruleitem* denote a *ruleitem* whose *condset* has *k* items.

Let *Fk* denote the set of frequent *k-ruleitems*.

Each element of this set is of the following form:

{(*condset*, *condsupCount*), (*y*, *rulesupCount*)}

Line 1-3 in figure 3.4 represents the first pass of the algorithm. It counts the item and class occurrences to determine the frequent 1-*ruleitems* (line 1). From this set of 1-*ruleitems*, a set of CARs (called *CAR*1) is generated by *genRules* (line 2).

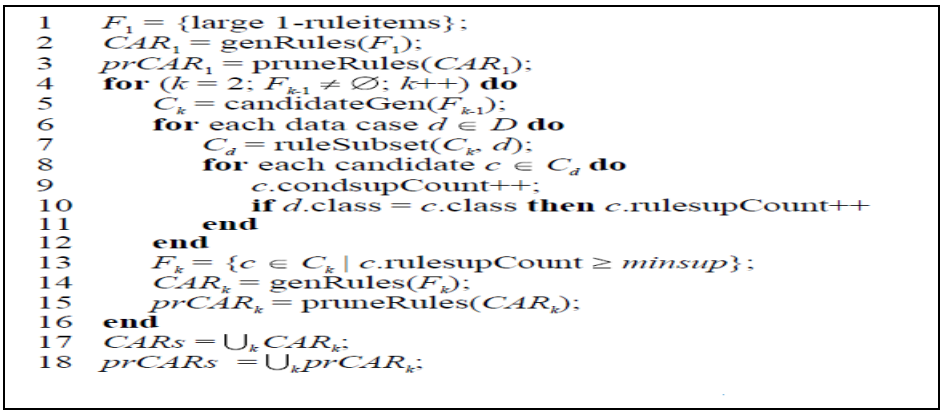


Figure 3.5: The CBA-CARs phase of CBA algorithm pseudo code

*CAR*1 is subjected to a pruning operation (line 3) Pruning is also done in each subsequent pass to *CARk* (line 15).

It obtained by deleting one condition from the conditions of *r*, then rule *r* is pruned. This pruning can cut down the number of rules generated substantially.

For each subsequent pass, say pass *k*, the algorithm performs 4 major operations.

First, the frequent *ruleitems Fk*-1 found in the (*k*-1)*th* pass are used to generate the candidate *ruleitems* C_k using the *condidateGen* function (line 5).

It then scans the database and updates various support counts of the candidates in *Ck* (line 6-12), after those new frequent *ruleitems* have been identified to form *Fk* (line 13).

The algorithm then produces the rules CAR_k using the *genRules* function (line 14).

Finally, rule pruning is performed (line 15) on these rules.

The *candidateGen* function is similar to the function *Apriori-gen* in algorithm Apriori. The *ruleSubset* function takes a set of candidate *ruleitems Ck* and a data case *d* to find all the *ruleitems* in *Ck* whose *condsets* are supported by *d*.

This and the operations in line 8-10 are also similar to those in algorithm Apriori. The difference is that we need to increment the support counts of the *condset* and the *ruleitem* separately whereas in algorithm Apriori only one count is updated. This allows us to compute the confidence of the *ruleitem*. They are also useful in rule pruning. The final set of class association rules is in *CARs* (line17). Those remaining rules after pruning are in *prCARs* (line 18) [27].

3.3.3 Class Aprior example

Finding class association rules mining requires reviewing the following terms

1- ruleitem represents transaction itemset labeled by class like this form ({condest},y) where condest represents itemsets in transaction vlues ,and y represents class values,

2- condsup represents how many times the condest is counted in transaction column.

3- rulesup represents how many times the ruleitem counted in table db.

4- Sup=(condsup/n) where n =number of documents.,conf=(rulesup/consup) .

No of Document	Transaction	Class
Doc1	Student, Teach, School	Education
Doc2	Student, School	Education
Doc3	Teach, School, City, Game	Education
Doc4	Baseball, Basketball	Sport
Doc5	Basketball, Player, Spectator	Sport
Doc6	Baseball, Coach, Game, Team	Sport
Doc7	Basketball, Team, City, Game	Sport

table 3.2. An example of a data set for mining class association rules

Every candidate ruleitem has minsup > sup threshold value is frequent and also has minconf > conf threshold value i.e it can be at the form of class association rules.

At this example minsup=0.15 , minconf=0.70.

At table 3.2 the generated 1-ruleitems is checked to determine frequrnt 1- ruleitems to form class association rules 1(CARs1).

ruleitem	condsup	rulesup	CARs	sup	conf
({condest},class)			Condest => class		
({Student},Ed)	2	2	{Student}=>Ed	2/7	2/2
({Teach},Ed)	2	2	{Teach}=>Ed	2/7	2/2
({School},Ed)	3	3	{School}=>Ed	3/7	3/3
({Game},Ed)	2	1	{Game}=>Ed	2/7	1/2
({City},Ed)	2	1	{City}=>Ed	2/7	1/2
({Baseball,Sport}	2	2	{Baseball}=>Sport	2/7	2/2
({Basketball},Sport)	3	3	{Basketball}=>Sport	3/7	3/3
({Player},Sport)	1	1	{Player}=>Sport	1/7	1/1
({Spectator },Sport)	1	1	{ Spectator}=>Sport	1/7	1/1
({Coach},Sport)	1	1	{Coach}=>Sport	1/7	1/1
({Team},Sport)	2	2	{Team}=>Sport	2/7	1/1
({Game},Sport)	3	2	{Game}=>Sport	3/7	2/3

Table3.4 Generating frequent 1-ruleitemset and class association rules 1

- Frequent 1- ruleitem ($F1$)

({School}, Education) [sup = 3/7]

({Student}, Education)}[sup=2/7]

({Teach}, Education}) [sup=2/7]

({Baseball}, Sport)[sup=2/7]

({Basketball},Sport) [sup=3/7]

({Game},Sport)[sup = 2/7]
({Team} , Sport) [sup = 2/7,]

- Class association rule 1(CARs1)

School => Education [sup = 3/7, conf = 3/3]
Student => Education [sup = 2/7, conf = 2/2]
Teach => Education [sup = 2/7, conf = 2/2]
Baseball => Sport [sup = 2/7, conf = 2/2]

Basketball => Sport [sup = 3/7, conf = 3/3]
Game =>Sport [sup = 2/7, conf = 2/3]
Team => Sport [sup = 2/7, conf = 2/2]

At table 3.4 the generated 2-ruleitems is checked to determine frequrnt 2- ruleitems to form class association rules 2(CARs2).

ruleitem	condsup	rulesup	CARs	sup	conf
({condest},class)			Condest => class		
({Student,Teach},Ed)	1	1	({Student,Teach}=>Ed	1/7	1/1
({Student,School},Ed)	2	2	{Student,School}=>Ed	2/7	2/2
({School,Teach},Ed)	2	2	{School,Teach}=>Ed	2/7	2/2
({Baseball, Basketball},Sport)	1	1	{Baseball, Basketball }=>Sport	1/7	1/1
({Basketball, Team},Sport)	1	1	{ Basketball, Team }=>Sport	1/7	1/1
({Basketball,Game},Sport)	1	1	({Basketball,Game}=>Sport	1/7	1/1
({Baseball, Team},Sport)	1	1	{Baseball, Team}=>Sport	1/7	1/1
({Baseball,Game},Sport)	1	1	{Baseball,Game}=>Sport	1/7	1/1
({Team,Game},Sport)	2	2	{Team,Game}=>Sport	2/7	2/2

Table3.5 Generating frequent 2-ruleitemset and class association rules 2

- Frequent 2- ruleitem (*F*2)

({School, Student}, Education)[2/7]
({School, Teach}, Education)[2/7]
({Game, Team}, Sport)[sup=2/7]

- Class association rule 2(CARs2)

School, Student => Education [sup = 2/7, conf = 2/2]
School, Teach => Education [sup = 2/7, conf = 2/2]
Game, Team => Sport [sup = 2/7, conf = 2/2]

3.4 Classification Based on Association Rules-Classifier Builder (CBA-CB) phase

In this section we present the algorithm for classification based on association algorithm for building a classifier using CARs. It will produce the best classifier out of the whole set

of rules would involve evaluating all the possible subsets of it on the training data and selecting the subset with the right rule sequence that gives the least number of errors.

Before presenting the algorithm, let us define a total order on the generated rules. This is used in selecting the rules for our classifier [31].

Basic concepts of building classifier:

Definition: Given two rules, *ri* and *rj*, $ri \succ rj$ (also called *ri* precedes *rj* or *ri* has a higher precedence than *rj*) if the following:

1. The confidence of *ri* is greater than that of *rj*.
2. Their confidences are the same, but the support of *ri* is greater than that of *rj*.
3. Both the confidences and supports of *ri* and *rj* are the same, but *ri* is generated earlier than *rj*.

Let *R* be the set of generated rules (i.e., CARs or pCARs), and *D* the training data. The basic idea of the algorithm is to choose a set of high precedence rules in *R* to cover *D*.

3.4.1 The Classifier based on CARs Algorithm:

Our classifier is of the following format:

<*r*1, *r*2, …, *rn*, *default_class*>,where $ri \in R$, $ra \succ rb$ if $b > a$.

In classifying an unseen case, the first rule that satisfies the case will classify it, If there is no rule that applies to the case, it takes on the default class as in building such a classifier is shown in Figure 3.5. It can be described into three steps:

Step 1 (line 1): Sort the set of generated rules *R* according to the relation " $\succ$ ",

This is to ensure that we will choose the highest precedence rules for our classifier.

Step 2 (line 2-13): Select rules for the classifier from *R* following the sorted sequence. For each rule *r*, we go through *D* to find those cases *covered* by *r* (they satisfy the conditions of *r*) (line 5).

We mark *r* if it correctly classifies *a* case *d* (line 6), *d*.id is the unique identification number of *d*, If *r* can correctly classify at least one case (i.e., if *r* is marked), it will be a potential rule in our classifier (line 7-8), those cases it covers are then removed from *D* (line 9).

```
1    R = sort(R);
2    for each rule r ∈ R in sequence do
3        temp = ∅;
4        for each case d ∈ D do
5            if d satisfies the conditions of r then
6                store d.id in temp and mark r if it correctly
                     classifies d;
7        if r is marked then
8            insert r at the end of C;
9            delete all the cases with the ids in temp from D;
10           selecting a default class for the current C;
11           compute the total number of errors of C;
12       end
13   end
14   Find the first rule p in C with the lowest total number
         of errors and drop all the rules after p in C;
15   Add the default class associated with p to end of C,
         and return C (our classifier).
```

Figure 3.6: the CBA-CB phase of CBA algorithm pseudo code

A default class is also selected (the majority class in the remaining data),which means that if we stop selecting more rules for our classifier *C* this class will be the default class of *C* (line10).

We then compute and record the total number of errors that are made by the current *C* and the default class (line 11).

This is the sum of the number of errors that have been made by all the selected rules in *C* and the number of errors to be made by the default class in the training data. When there is no rule or no training case left, the rule selection process is completed.

Step 3 (line 14-15): Discard those rules in *C* that do not improve the accuracy of the classifier. The first rule at which there is the least number of errors recorded on *D* is the cutoff rule.

All the rules after this rule can be discarded because they only produce more errors. The undiscarded rules and the default class of the last rule in *C* form our classifier [27].

Chapter 4
Implementation

Chapter 4

4.1 Introduction

In this chapter we will use some programming languages and tools to implement our proposed system. These programming languages are illustrated as follows:

1. C#: C# is an interpreter, high-level programming language, C# is used as it is the main programming language for this work, also C# is involved practically in most parts of the system,The C# interface is shown below 4.1.

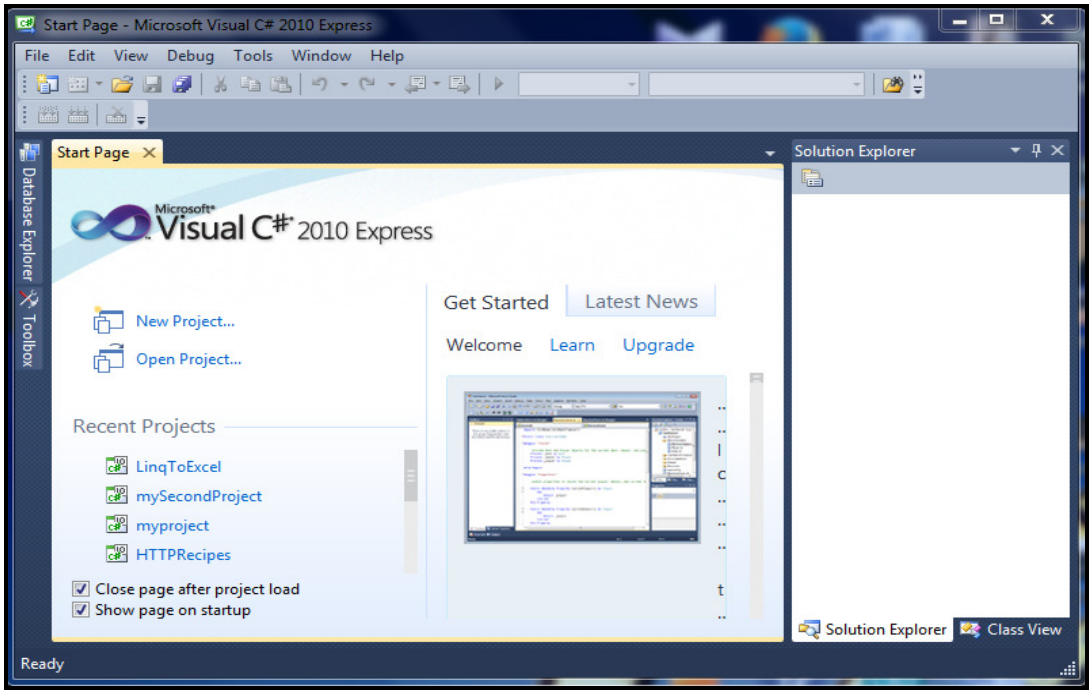


Figure 4.1: Visual C# main interface

2. **Software Tools and Frameworks**: We will use a variety of readily-available software tools and frameworks to deal with the incidental tasks of software development and be able to concentrate on the main objectives of this research. These tools are listed as follows:

    a. LINQ: **Language Integrated Query** is a Microsoft .NET Framework component that adds native data querying capabilities to .NET languages.

    b. **Microsoft *Excel***: is a spreadsheet program which allows us to create log flat files holding all of the weblogs files collecting for different platforms as shown below in figure 4.2.

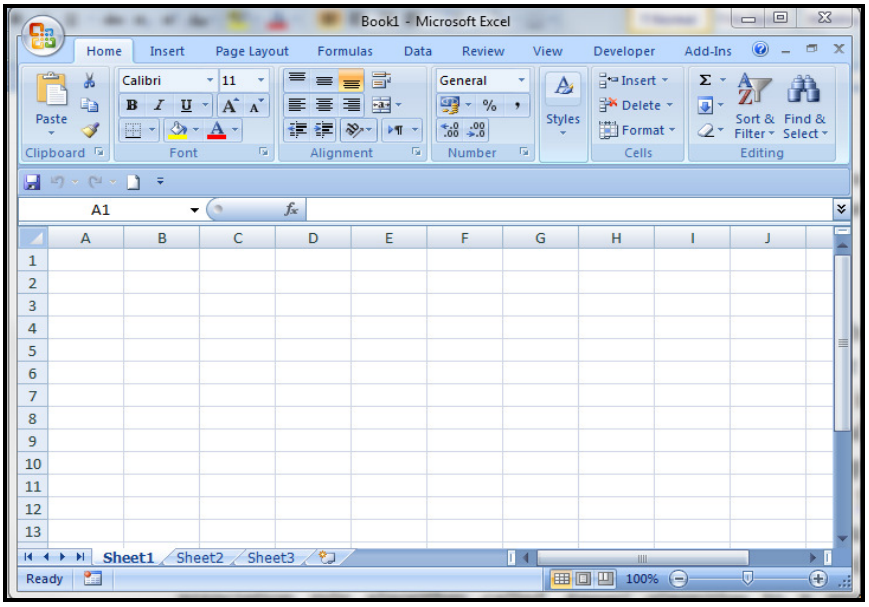


Figure 4.2: Excel file spreadsheet

4.2 System Workloads

4.1.1 Prepare the Simulated Log-file

We used Excel file with extension xlsx to create a flat file to hold the content of the log-file as shown below in figure 4.3.

	A	B	C	D
1	TimeDate	URL	User	Type
2	05:10 1/17/2013	http://www.databaseanswers.org/index.htm	pc1	Education
3	05:13 1/17/2013	http://www.dotnetperls.com	pc1	Education
4	05:17 1/17/2013	http://www.dotnetperls.com/datacolumn	pc1	Education
5	05:22 1/17/2013	http://databases.about.com/od/specificproducts/a/firstnormal	pc1	Education
6	05:26 1/17/2013	http://www.java2s.com	pc1	Education
7	05:30 1/17/2013	http://www.java2s.com/Code/CSharp/Database-ADO.net/Getta	pc1	Education
8	07:30 1/17/2013	http://www.ferryhalim.com/orisinal/	pc2	Entertainment
9	07:33 1/17/2013	http://www.ferryhalim.com/orisinal/g3/carrot.htm	pc2	Entertainment
10	07:40 1/17/2013	http://www.complete-review.com/	pc2	Entertainment
11	07:42 1/17/2013	http://www.complete-review.com/main/main.html	pc2	Entertainment
12	07:43 1/17/2013	http://www.aljazeera.net/news/arabic	pc2	News
13	07:44 1/17/2013	http://www.aljazeera.net/news/pages/db6f5a3f-e19c-4471-866	pc2	News
14	07:48 1/17/2013	http://www.aljazeera.net/news/pages/7a4ecdc4-13ad-4a63-bf8	pc2	News
15	07:50 1/17/2013	http://www.aljazeera.net/news/pages/90d1c951-7982-4905-849	pc2	News
16	07:56 1/17/2013	http://www.aforgenet.com/	pc1	Education
17	08:00 1/17/2013	http://www.aforgenet.com/framework/docs/	pc1	Education
18	08:07 1/17/2013	http://www.alarabiya.net/default.html	pc2	News
19	08:09 1/17/2013	http://www.alarabiya.net/articles/2013/01/17/261048.html	pc2	News

Figure 4.3: a simulated log file

4.2.2 Connecting C# to Log-File Table

We used C# programming language code to import log-file table to our implemented proposed application by using Linq-To-Excel in the 'dot' Net library media. The Linq-To-Excel library beside its capability to facilitate connecting Excel spreadsheets it can also be used to query Excel spreadsheets using the LINQ syntax. To add the library to the application the library can be referenced to the solution explore window as illustrated in the below figure4.4.

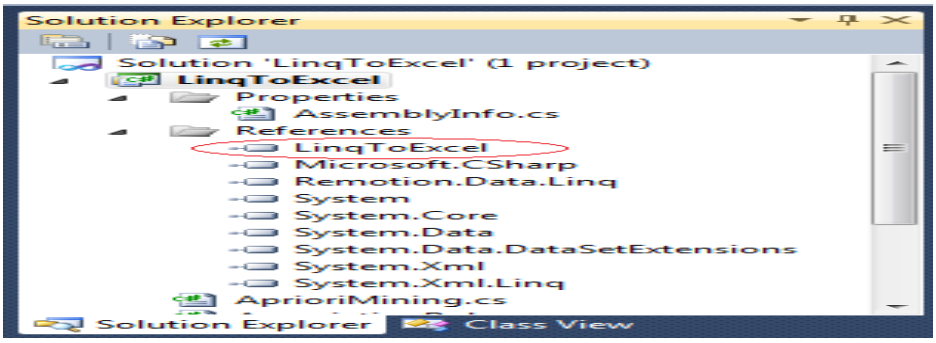


Figure 4.4: Adding Library to the Solution Explorer

The library is browsed after downloading by adding the following as shown below in figure 4.5:

- LinqToExcel.dll.
- Remotion.Data.Linq.dll

This library is downloaded from: http://code.google.com/p/linqtoexcel

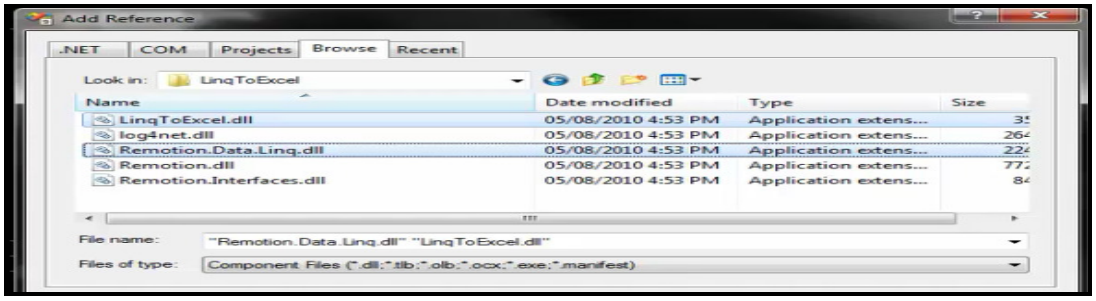


Figure 4.5: Browsing to library references

Using Linq-To-Excel library in C# programming editor as the following steps to connect and import the excel log file table to our implemented program:

1. Creating log-file class: we create a class called log file to represent the columns of the simulated log file as shown below in figure 4.6.

```
namespace LinqToExcel
{
    class LogFile
    {
        public DateTime Date { get; set;}
        public string   URL { get; set;}
        public string   User { get; set;}
        public string   Type { get; set;}
```

Fig 4.6: creating the Logfile class

2.Importing log-file content: we write function to import and display the contents of simulated log-file as shown below in figure 4.7.

```
public static void DisplayLogFile()
{

    var excel = new ExcelQueryFactory();
    excel.FileName = @"E:\LogFile.xlsx";
    var dataset = from data in excel.Worksheet<LogFile>()
                  select data;
    foreach (var item in dataset)
    {
        Console.WriteLine("\nDate: {0}  URL:{1} User:{2} Type:{3}", item.Date, item.URL, item.User, item.Type);
    }

}
```

Figure 4.7: function to connect and display log-file.

The imported database of log file as shown below in *figure 4.8.*

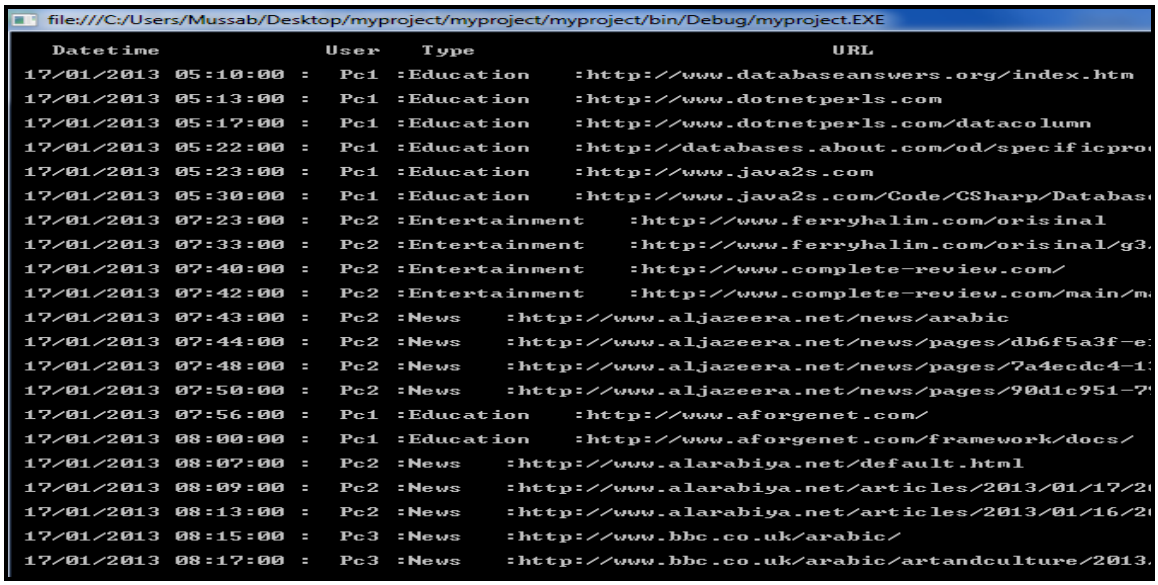


Figure 4.8:Import a sample of log-file

3.Creating sessions :we create sessions for every user and turn it into a transactional database as shown below in figure 4.9:

```
public static ItemsetCollection ProduceAllDurationLists(List<DateTime> DurationList, int index, string USER)
{
    TimeSpan DurationDiffrence;
    int nextIndex = index;

    var AllDurations = new List<List<DateTime>>();
    var AllTheUniqueSitesType = new ItemsetCollection();
    while (DurationList.Count > index)
    {
        var FirstDuration = DurationList[index];
        var Duration = new List<DateTime>();
        var SitesType = new Itemset();
        var UniqueSitesType = new Itemset();
        for (int i = index; i < DurationList.Count; i++)
        {

            DurationDiffrence = DurationList[i].Subtract(FirstDuration);
            if (DurationDiffrence.TotalMinutes < 30)
            {
                Duration.Add(DurationList[i]);
                nextIndex++;
            }
            else { break; }
        }
        index = nextIndex;
       // SitesType = LogFile.GetSiteTypeCorrespondingToDateTime( Duration, USER);
        SitesType = LogFile.CoorespondingOfDateTimeAndType(Duration,USER);
        UniqueSitesType = LogFile.GetuniqueType(SitesType);
        AllTheUniqueSitesType.Add(UniqueSitesType);
        AllDurations.Add(Duration);
    }
    return AllTheUniqueSitesType;
}
```

Figure 4.9 function to convert sessions to transactional database

The generated transactional database of log file is shown below where every line represents a user's session as shown below in figure 4.10:

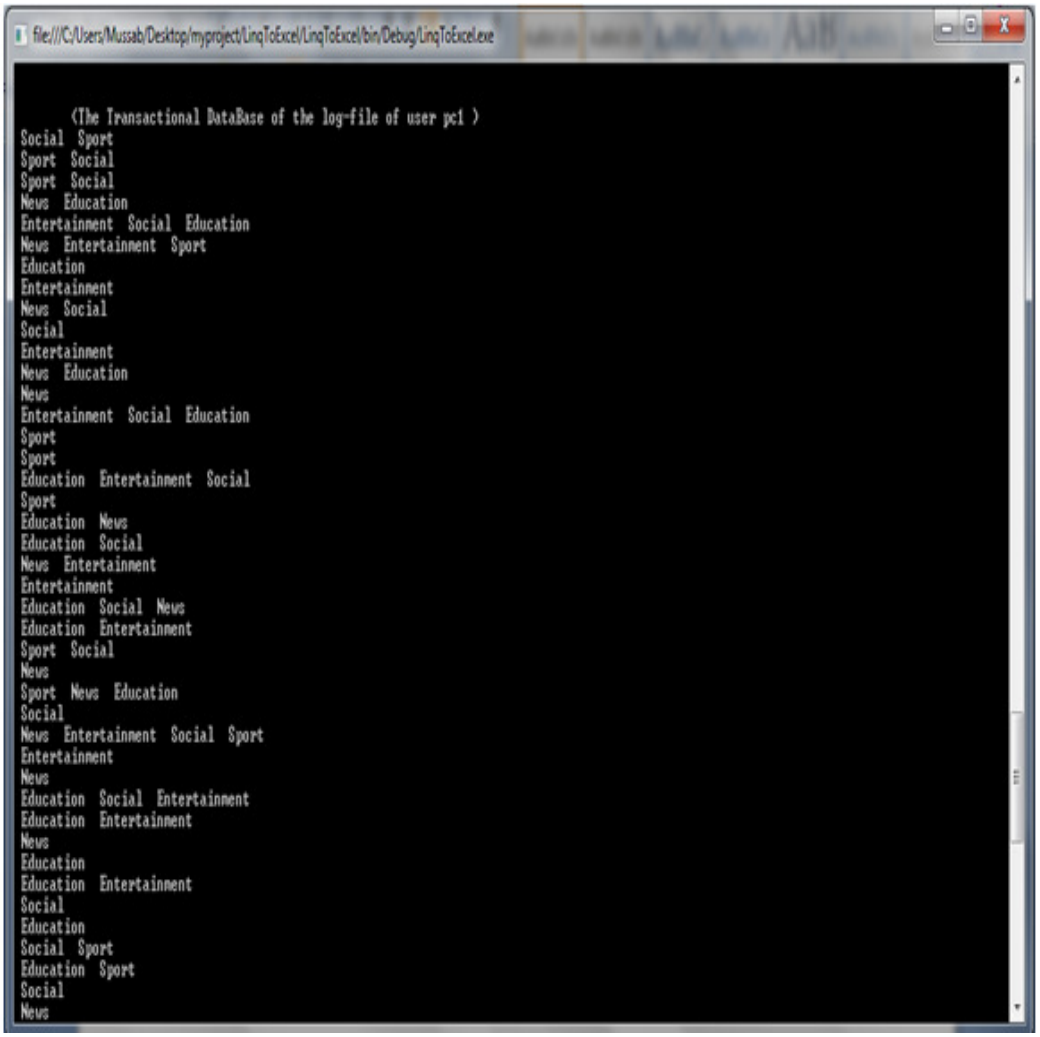


Figure 4.10:A Sample of transactional database

4.3 Applying Aprior algorithm

The algorithm is based on the following steps:

1-Finding the frequent itemsets :They are performed from generated candidate itemsets that has a minimum support threshold above 0.5. The function code of finding the candidate & frequent itemsets are illustrated in the follwing figure 4.11:

```
public static ItemsetCollection DoApriori(ItemsetCollection db, double supportThreshold)
{   Itemset I = db.GetUniqueItems();
    ItemsetCollection L = new ItemsetCollection(); //resultant large itemsets
    ItemsetCollection Li = new ItemsetCollection(); //large itemset in each iteration
    ItemsetCollection Ci = new ItemsetCollection(); //pruned itemset in each iteration
                int i = 1;
                //first iteration (1-item itemsets)
    foreach (string item in I)
    {
        Ci.Add(new Itemset() { item });
    }
    //next iterations
    int k = 2;
    while (Ci.Count != 0)
    {
        //set Li from Ci (pruning)

        Li.Clear();
        foreach (Itemset itemset in Ci)
        {
            itemset.Support = db.FindSupport(itemset);
            if (itemset.Support >= supportThreshold)
            {
                Li.Add(itemset);
                L.Add(itemset);
            }
        }                                                   //set Ci for next iteration (find supersets of Li)
        Ci.Clear();
        Ci.AddRange(Bit.FindSubsets(Li.GetUniqueItems(), k)); //get k-item subsets
        i += 1;
        k += 1;
    }
    return (L);
}
```

Figure 4.11: function of finding frequent itemsets of aprior algorithm.

The functionality of finding the candidate & frequent itemsets is that the aprior algorithm scans the number of times the emergence of 1-itemset in the database to calculate the support for each candidate itemset. It then compares it with the support threshold. If the itemset support is bigger than the support threshold then the candidate itemset will be added to the frequent itemset as shown in figure 4.12 below.

```
The candiate 1-itemset

{Social} (support: 37.5%)
{Sport} (support: 31.25%)
{News} (support: 36.25%)
{Education} (support: 33.75%)
{Entertainment} (support: 38.75%)

The frequent 1-itemset

{Social} (support: 37.5%)
{Sport} (support: 31.25%)
{News} (support: 36.25%)
{Education} (support: 33.75%)
{Entertainment} (support: 38.75%)
```

Figure 4.12: Shows a sample result of frequent 1-itemset after applying support threshold on candidate 1-itemset .

In the next figure the three first candidate itemsets has support less than the support threshold so these itemset are not added to the frequent itemsets as shown in figure 4.13 below.

```
                The candiate 4-itemset

    {Sport, Social, Entertainment, Education}
    {News, Social, Entertainment, Education}
    {News, Sport, Entertainment, Education}
    {News, Sport, Social, Education} (support: 1.25%)
    {News, Sport, Social, Entertainment} (support: 2.5%)

                The frequent 4-itemset

    {News, Sport, Social, Education} (support: 1.25%)
    {News, Sport, Social, Entertainment} (support: 2.5%)
```

Figure 4.13: shows a sample result of frequent 4-itemset after applying support threshold on candidate 4-itemset .

The only candidate itemsets that appears in the below figure 4.14 that has support less than the support threshold, consequently the frequent itemset is not generated.

```
                The candiate 5-itemset

    {News, Sport, Social, Education, Entertainment}

                The frequent 5-itemset
```

Figure 4.14: shows a sample result of frequent 5-itemset after applying support threshold on candidate 5-itemset.

2-Finding the association rules: That is accomplished from the final frequent itemsets with a confidence threshhold above 80.00 .

The function code of finding association rules by using the resulted final frequent itemset is shown in the figure 4.15 below.

```
public static List<AssociationRule> Mine(ItemsetCollection db, ItemsetCollection L, double confidenceThreshold)
{
    List<AssociationRule> allRules = new List<AssociationRule>();
    foreach (Itemset itemset in L)
    {
        ItemsetCollection subsets = Bit.FindSubsets(itemset, 0); //get all subsets
        foreach (Itemset subset in subsets)
        {
            double confidence = (db.FindSupport(itemset) / db.FindSupport(subset)) * 100.0;
            if (confidence >= confidenceThreshold)
            {
                AssociationRule rule = new AssociationRule();
                rule.X.AddRange(subset);
                rule.Y.AddRange(itemset.Remove(subset));
                rule.Support = db.FindSupport(itemset);
                rule.Confidence = confidence;
                if (rule.X.Count > 0 && rule.Y.Count > 0)
                {
                    allRules.Add(rule);
                }
            }
        }
    }
    return (allRules);
}
```

Figure 4.15 function code of applying association rules .

Applying the support threshold for every possible generated itemsets, with the last satisfied frequent itemsets are used to apply association rules as shown in figure 4.16.

.

Figure 4.16: shows a sample result of the frequent itemsets ready for association rules.

To find the association rules of the itemset as {News , Sport , Social ,Education} as follows

A. The possible nonempty generated subitemsets generated of previous itemset are

{Education},{Social},{Social, Education},{Sport},{Sport, Education},

{Sport, Social},{Sport, Social, Education},{News},{News, Education}

{News, Social},{News, Social, Education},{News, Sport},

{News, Sport, Education} and {News, Sport, Social}.

B. every nonempty subsetitemset of itemset, we can extract the strong rule in the next form as follows:

{subitemset}➔{itemset - subitemset}

Where *confidence =(sup-count(itemset)/sup-count(subitemset)) >= 80%*

Satisfying the previous formula on the generated subitemsets of the itemset {Education, News, Social, Sport}, it will produced the following results:

{Education}➔ {Social,News,Sport} , Confidence=3.70 %

{Social}➔ {Education,News,Sport} , Confidence=3.33%

{Social, Education}➔{News,Sport} ,Confidence=11.11%

{Sport}➔{News,Social,Education} , Confidence=4.00%

{Sport, Education}➔{News,Social} ,Confidence=25.00%

{Sport, Social}➔{Education,News} ,Confidence=10.00%

{Sport, Social, Education}➔{News} ,Confidence=100%

{News}➔{Sport,Social,Education} ,Confidence=3.44%

{News, Education}➔{Social,Sport} ,Confidence=10%

{News, Social}➔{Education,Sport} ,Confidence=14.28%

{News, Social, Education}➔{News} ,Confidence=33.33%

{News, Sport}➔{Education,Social} ,Confidence=14.28%

{News, Sport, Education}➔{Social} ,Confidence=50.00%

{News, Sport, Social}➔{Education} ,Confidence=33.33%

Every rule has confidence less than 80% is discarded.

We also can find the association rules for the itemset {New, Social, Sport, Entertainment} by applying the same previous steps which can generates the classes and nominate the final resulting association rules in the frequent itemset. below figure 4.17 expresses more:

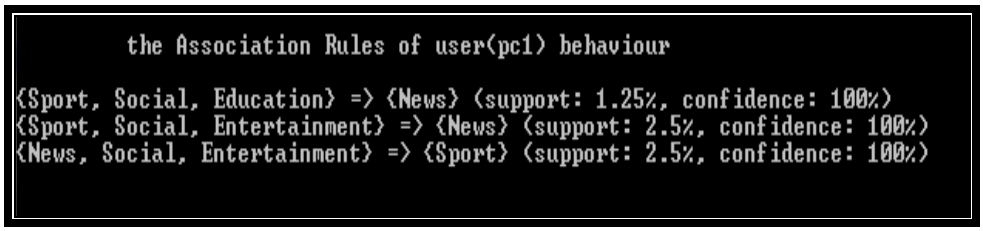


Figure 4.17: the result of association rules of the final frequent itemsets

4.3 Applying Classification Based on Association Algorithm (CBA) Algorithm

Previously in aprior algorithm sample result of association rules, we've found a pattern describes the surfing from category (class page type) to another, but not able to classify the user behavior. In this case we've applied CBA algorithm which can integrate classification with association which is applied in two steps.

A-Generating CBA-To- CARs .

By setting the association rules in the pages content at the right hand side (r.h.s) and the class page type at the left hand side (l.h.s), were it had achieved by the writing code function.

- The function code of finding the candidate & frequent itemsets for generating CARs are illustrated as shown below 4.18:

```
public static ItemsetCollection CBA_CARs(ItemsetCollection TransactionDataBase, ItemsetCollection Class, double support, double confidence)
{
    Itemset I = TransactionDataBase.GetUniqueItems();
    ItemsetCollection Ci = new ItemsetCollection();
    ItemsetCollection Li = new ItemsetCollection(); //large itemset in each iteration
    ItemsetCollection L = new ItemsetCollection(); //resultant large itemsets
     foreach (var item in I){Ci.Add(new Itemset { item }); Console.WriteLine(item);
    }
    int k = 2, condSupCount = 0, ruleSupCount = 0;
    string[] Matchedclass = { " " };
    string[] condest = { "" };
    int A = 1;
    do
```

```
        {   Li.Clear();
            foreach (var itemset in Ci)
            {
                for (int i = 0; i < TransactionDataBase.Count && i < Class.Count; i++)
                {
                    if (TransactionDataBase[i].Contains(itemset))
                    {
                        condSupCount++;
if (Class[i].Contains("Education") || Class[i].Contains("Sport") || Class[i].Contains("Entertainment") || Class[i].Contains("News") || Class[i].Contains("Social"))
                        {
                            ruleSupCount++;
                            Matchedclass = Class[i].ToArray();
                            condest = itemset.ToArray();
                        }
                    }
                }
                itemset.Support = ((double)condSupCount / (double)TransactionDataBase.Count) * 100.00;
                itemset.Confidence = ((double)ruleSupCount / (double)condSupCount) * 100.00;
                itemset.MatchedClass = Matchedclass;
                itemset.condest = condest;
                if (itemset.Support >= support && itemset.Confidence >= confidence)
                {
                    Li.Add(itemset);
                    L.Add(itemset);
                }
                condSupCount = 0;
                ruleSupCount = 0;
                }
            Ci.Clear();
            var getKitemSubsets = Bit.FindSubsets(Li.GetUniqueItems(), k);
            Ci.AddRange(getKitemSubsets); //get k-item subsets
            k++;
            A++;
        } while (Ci.Count > 0);
        return L;
```

Figure 4.18 function code of finding frequent itemsets for generating CBA-To- CARs .

CARs can be determined by calculating three following terms:

I. condSupCount : number of counting the itemset in transaction database.

II. ruleSupCount : number of counting the itemset transaction database and is labled with classType.

III. Confidence for every rule

$$confidence = (ruleSupCount / condSupCount) .$$

Then we can extract CARs for candidate-1 itemset as shown in figure 4.19 below by setting Support threshold =1.5,confidence 90.

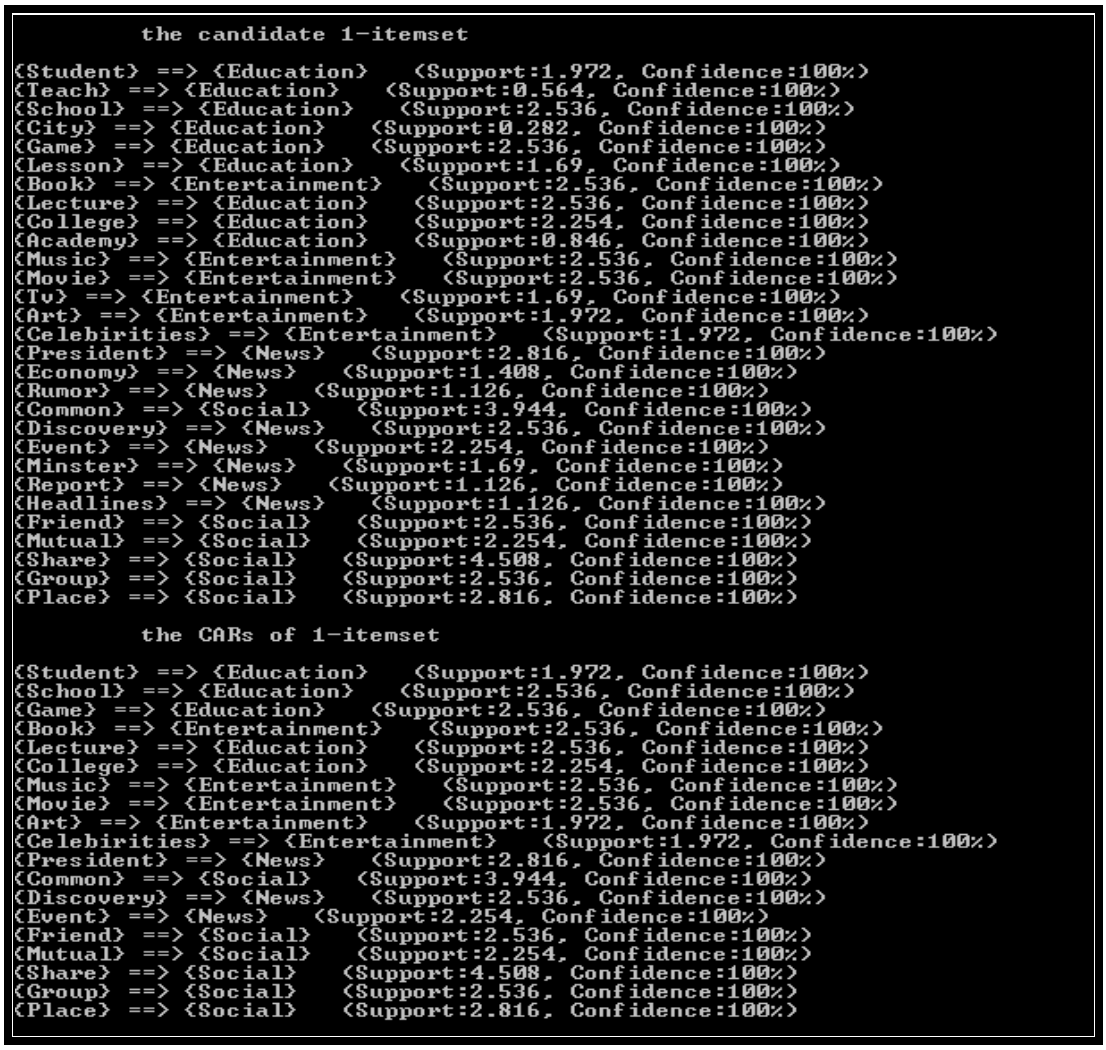


Figure 4.19: shows CARs sample result of candidate 1-itemset .

Sequentially we gather all the frequent CARs as shown in the next figure 4.21 below.

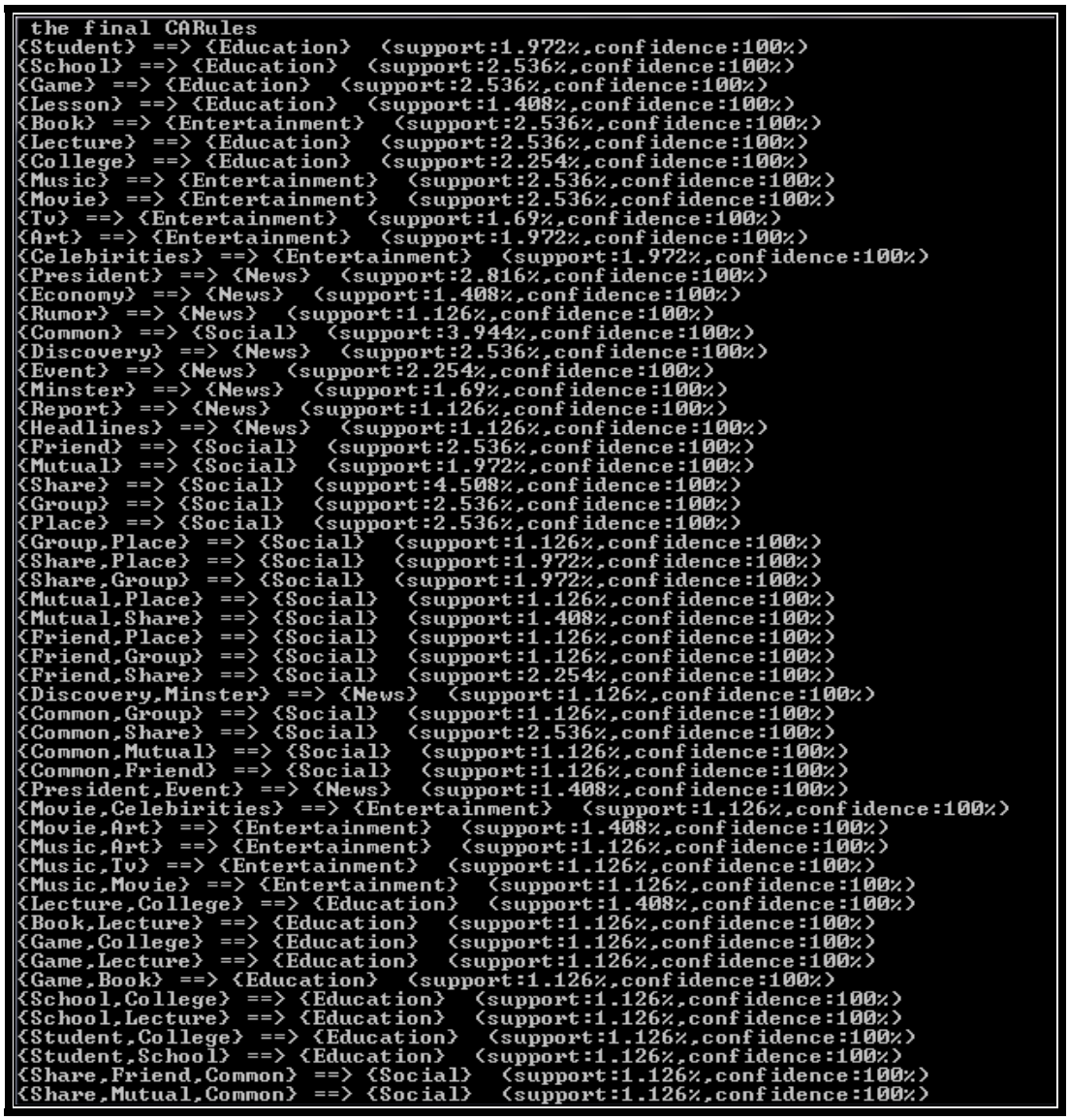


Figure 4.20 shows a sample result of the generated CARs.

2.Building CBA-CB

After generating CARs, building the classifier can be done in the next following steps.

I. Sort CARs in descending order according to support & confidence.

II. Initiate a classifier (*C*).

III. Pass the ordered CARs (*R*) through Database (*D*).

IV. Find the rules of *R* that covers *D* then put the covered rules at the end of *C*.

V. Remove all the matched cases in *D*.

VI. Select the default class of the class in the remaining *D*.

VII. Compute the total numbers of errors to evaluate the classifier.

VIII. Returning the Classifier.

The function code of CBA-CB execute the previous steps is shown the figure 4.21

```
public static ItemsetCollection CBA_CB(ItemsetCollection L,ItemsetCollection TransactionDataBase  ,ItemsetCollection Class)
{
    IEnumerable<Itemset>  orderList = L.OrderByDescending(l => l.Support);//order CARs
    var OrderRules = new ItemsetCollection();
    OrderRules.AddRange(orderList);

    var Classifier = new ItemsetCollection();
    var IndexOfD = new List<int>();
    var MarkedIndexOfR = new List<int>();
    IEnumerable<string> QueryCondest;
    IEnumerable<string> QueryClass;
    string[] newCondest = new string[] { "" };
    string[] newClass = new string[] { "" };
    string defaultClass = "";
    for (int j = 0; j < OrderRules.Count(); j++)
    {
        IndexOfD.Clear();
        MarkedIndexOfR.Clear();
        for (int i = 0; i < TransactionDataBase.Count && i < Class.Count; i++)
        {
            QueryCondest = TransactionDataBase[i].Intersect(OrderRules[j].condest);
            newCondest = QueryCondest.ToArray();

            if (newCondest.Length != 0)
            {
                QueryClass = Class[i].Intersect(OrderRules[j].MatchedClass);
                newClass = QueryClass.ToArray();
                if (newClass.Length != 0)
                {
                    IndexOfD.Add(i);
                    MarkedIndexOfR.Add(j);
                }
                else
                {
                    Console.WriteLine("Recording Error");
```

```
                }
            }

        }
        if (IndexOfD.Count != 0)
        {   IndexOfD.Reverse();
            Console.WriteLine();
             foreach (var item in IndexOfD)
            {
                TransactionDataBase.RemoveAt(item);
                Class.RemoveAt(item);
            }
             if (Class.Count != 0)
             {
                 defaultClass = Class.GetUniqueItems().First();
                 OrderRules[j].DefaulClass = defaultClass;
                 Classifier.Add(OrderRules[j]);
                 Classifier.Reverse();
                 Console.WriteLine();
             }
             else { break;}

        }
        for (int i = 0; i < TransactionDataBase.Count && i < Class.Count; i++)
        {Console.WriteLine("\n{" + String.Join(",", TransactionDataBase[i]) + "}" + " ==> " + "{" + String.Join("", Class[i]) + "}");}
    }
    Console.WriteLine();
    foreach (var item in Classifier)
    {
        Console.WriteLine("\n{" + String.Join(",", item.condest) + "}" + " ==> " + "{" + String.Join("", item.MatchedClass.ToArray()) + "}" + "\t( DefaulCla
    }

    Console.WriteLine();
    return Classifier;
}
```

Figure 4.21: function code of building classifier CBA-CB .

The classifier produces the most access pages and their categories as shown in figure 4.23

Figure 4.22:The classifier result of the CBA-CB

Eventually, after a classifier is constructed, it needs to be evaluated for

accuracy = (100 –(number of recoded error/number of database records))*100 .

Extracting the default class which is the majority class of the remaining database in figure 4.24 is shown below.

Figure 4.23: The classifier accuracy

Chapter 5
Results and Conclusions

Chapter 5

5.1 Introduction

This chapter we reviews all stages of building a predictive model for understanding user behavior.

Understanding user behavior online has become a requirement for large companies to commercialize products, this has become a need to build applications and rebuilding websites and engines recommendation in manner that meets the needs of the user. Understanding the behavior of the user must discover the navigating patterns; through the analysis of the clicking stream of users from page to another to build a predictive model generates output considered as parameters input to the recommendation engine to bring out the recommendations reflect the requirements of the user.

5.2The Results

In this work results are visualized and illustrated the obtained from the predictive model of six users, in two different forms:

1- Illustrating the classification results which contains the classes of the most accessed pages ,the accuracy of used classifier and the default class, in addition visualizing pie charts that illustrate the ratios used to browse the following class pages {News, Education, Entertainment, Social} as follows:

- Showing the classification results of the most accessed pages and visualizing User1 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.1.

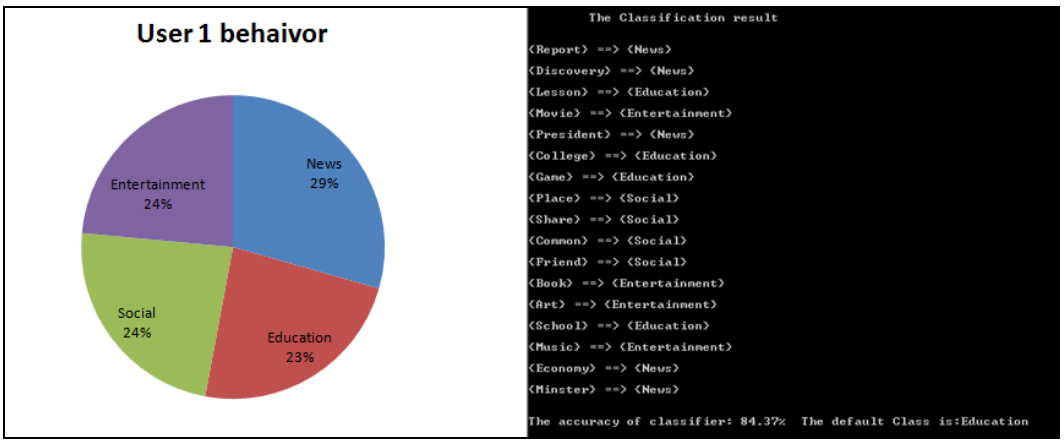


Figure 5.1: shows results visualizes user1 behavior

- Viewing the classification results of the most accessed pages and visualizing User2 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.2.

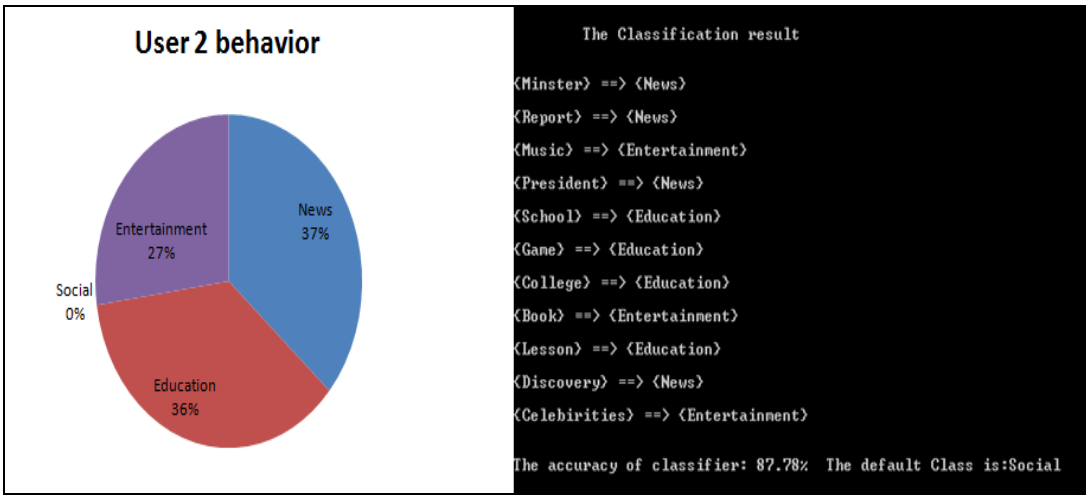


Figure 5.2: shows results visualizes user2 behavior

- The presentation of classification results of the most accessed pages and visualizing User3 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.3

.

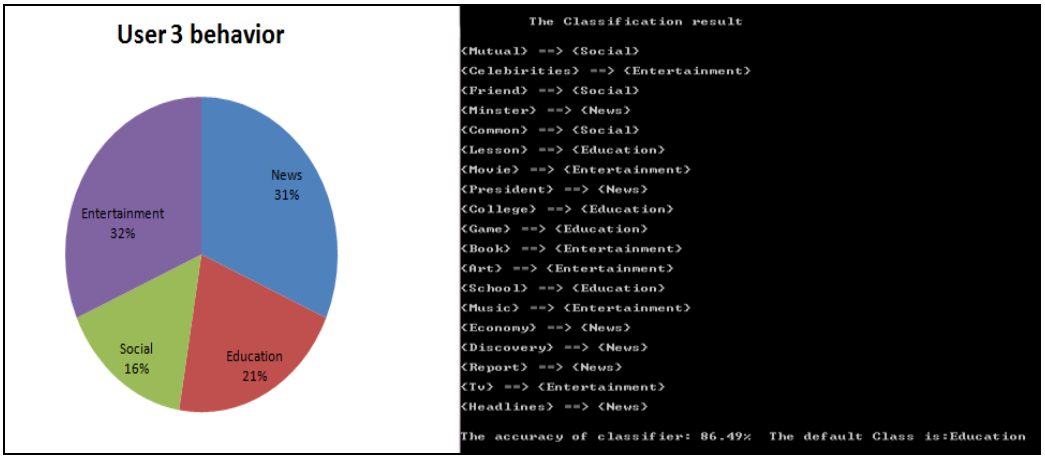


Figure 5.3 shows results visualizes user3 behavior

- Showing the classification results of the most accessed pages and visualizing User4 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.4

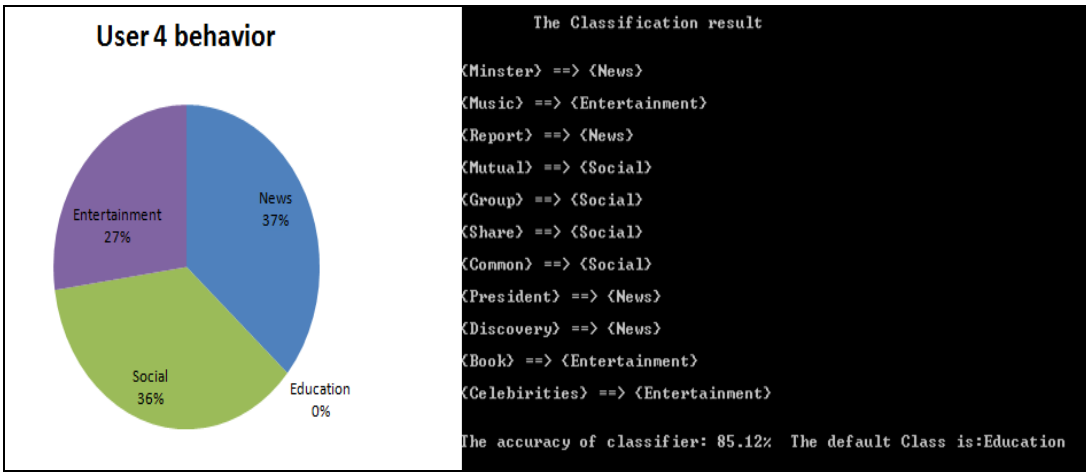


Figure 5.4 shows results visualizes user4 behavior

- Viewing the classification results of the most accessed pages and visualizing User5 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.5

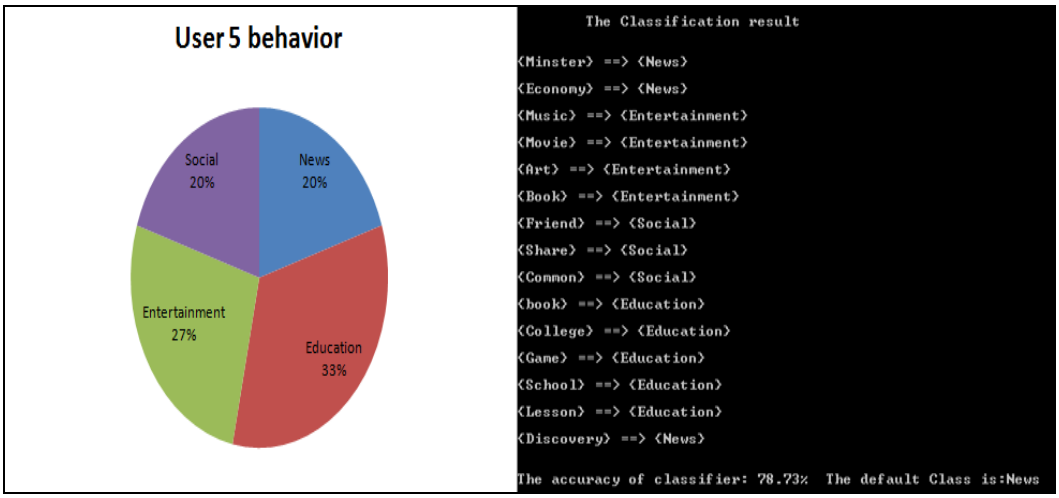


Figure 5.5 shows results visualizes user5 behavior

- Illustrating the classification results of the most accessed pages and visualizing User6 behavior in a shape of pie chart represents the ratio of browsing between different classes of accessed pages in figure 5.6.

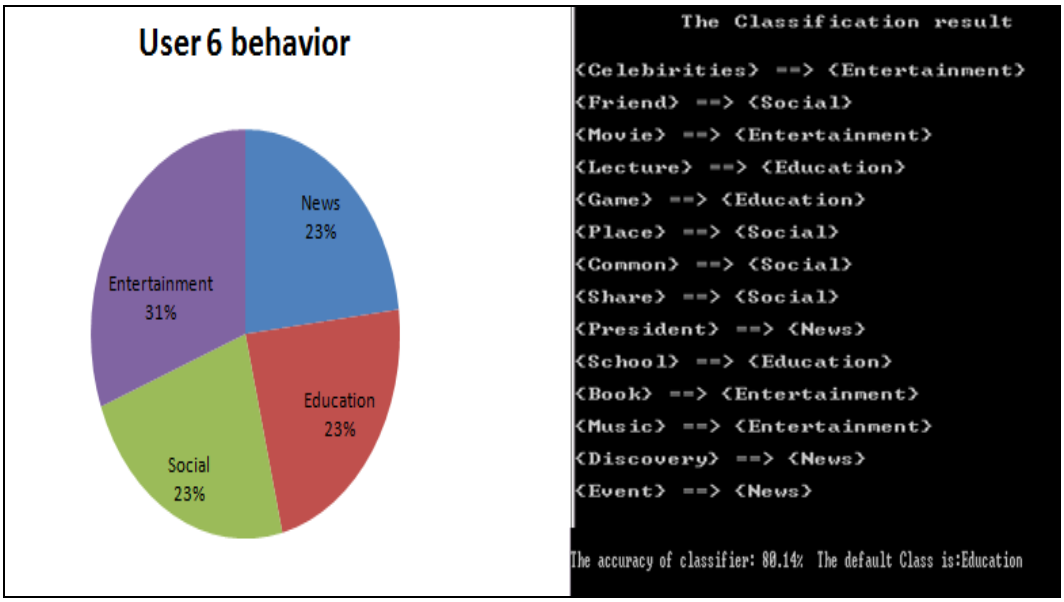
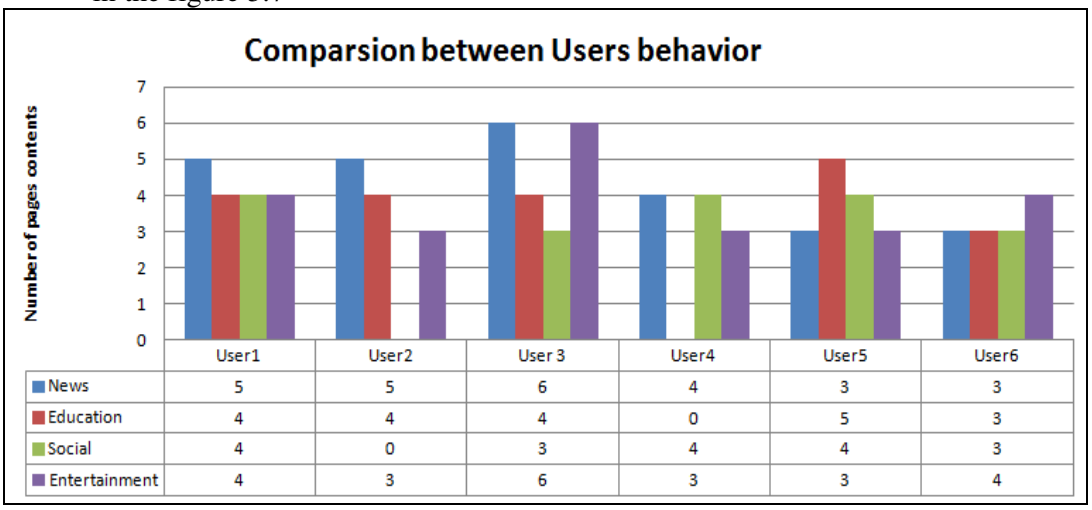


Figure 5.6 shows results visualizes user6 behavior

2- Demonstrating comparison of columns charts for each user are shown below in the figure 5.7



	User1	User2	User 3	User4	User5	User6
News	5	5	6	4	3	3
Education	4	4	4	0	5	3
Social	4	0	3	4	4	3
Entertainment	4	3	6	3	3	4

Figure 5.7 shows a chart comparison of different users .

A comparison of users according to pages class is indicated also in figure 5.8.

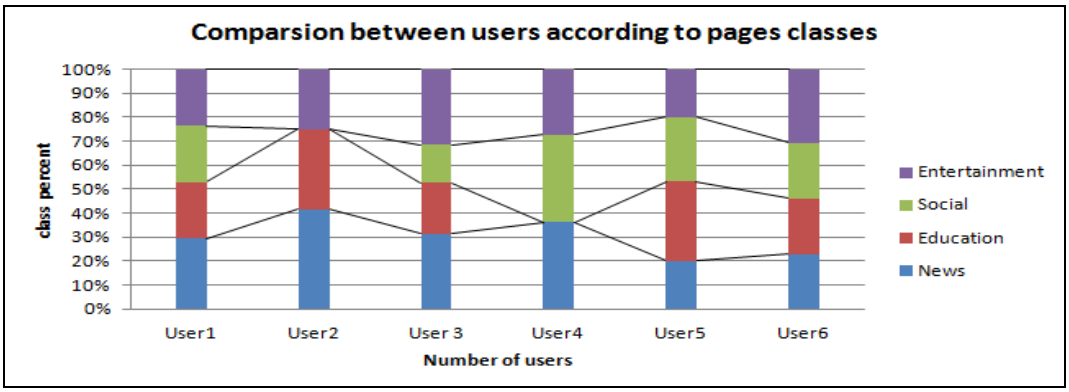


Figure 5.8 shows a chart comparison of different users .

5.3 Conclusion

In this thesis the proposed system receives its input by preprocessing data from log-file which is considered as a data source of user profile, we used in the analysis of a program a flat file in Excel format. time and date are extracted , and merged them into a single column, and the derivation of the content of the page was in different ways, either to extract meta-tag that found in the HTTP Header, or it can be extracted through the pages path in the URL, or by using meta-tag extractor applications, additional column holding the URL will be added as well as a new column specifies the class type of every page , and also a column holds the IP address is extracted to determine the user. Through all these columns we can extract sessions for each group of pages. Therefore, the transactional database can be created for every user.

After the transactional database is created, it is passed as parameter function to CBA algorithm, which consists of two phases. The first phase using the association rules to produce co-occurrence relationships, called associations among pages, this could be done using normal association rule in ordinary transaction database represented in aprior algorithm which is capable of describing and discovering patterns(which is resulted of the frequent item sets) ,but it doesn't have property to link every item set to its predefined class type(category),in case we use class association rules represented in class aprior algorithm ,it is like aprior algorithm ,capable of finding the frequent item sets, but unlike the aprior algorithm ,it has property which is able to link every item set to predefined class type ,also can count occurrences of every item set and its class type in transactional database, in turn generating classed association rules, which are used as part of the next phase ,the classifier building phase.

At the second phase of building the classifier ,the generated class association rules are resorted the rules according to support & confidence values ,then the ordered rules are passed through the transactional database to discover the matched rules, after that the classifier is initiated and the matched rules are added to the classifier, then the matched rules are removed of the transactional database ,select the default class of the major

remaining classes in transactional database ,we evaluate the classifier by counting the total number of errors divided by the number of tested data, finally we have a classifier contains the most accessed pages and their categories to form a user profile represents the user behavior.

5.4 Future Work

In this thesis we only explored a small field in using the CBA algorithm approach in understanding user behavior. There are many possible directions I wish to work in future. These directions include the following fields:

- Applying algorithm Classification based on Multiple Association Rule (CMAR), instead of CBA, in the case of CPMR uses support with multi-minimum support instead of single minimum support for improving accuracy.

- Integrating class association rules with different neural network classification algorithms.

- Applying sequential pattern techniques.

References:

[1] Bing Liu,"Web data mining ",2nd edition, 2011.

[2] A. J. Ratnakumar, "An Implementation of Web Personalization Using Web Mining Techniques," Journal of Theoretical and applied information technology, 2005.

[3] M.Srividya, D.Anandhi M.S.Irfan Ahmed, "Web Mining and Its Categories", International Journal Of Engineering And Computer Science ISSN:2319-7242

[4] Jiawei Han,Micheline Kamber,Jian Pei," Data Mining Concepts and Techniques", third edition , 2012 by Elsevier Inc.

[5] Ganesh Dhar, Govind Murari Upadhyay , Web Mining: Concepts and Decision Making Aid,International Journal of Advanced Research in Computer Science and Software Engineering ,IITM, Jnakpuri New Delhi, India

[6] J. Srivastava, R. Cooley, M. Deshpande, P. N. Tan. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. SIGKDD Explorations,2003.

[7] Kun-lung Wu, Philip S Yu, and Allen Ballman. Speed- tracer: A web usage mining and analysis tool. IBM Systems Journal, 37(1), 2006.

[8] Belsare Satish and Patil Sunil, Study and Evaluation of user's behavior in e-commerce Using Data Mining, Research Journal of Recent Sciences ISSN 2277 - 2502 Vol. 1, 375-387 (2012), Department of Computer Science, SCMIPS, Indore, MP, INDIA.

[9] E Hawkins, D.I., Best, R.J. and Coney, K.A., 2001, Consumer behavior: Building Marketing Strategy. New York: McGraw-Hill/Irwin 9th Edition.

[10] Na Li and Ping Zhang, "CONSUMER ONLINE SHOPPING ATTITUDES AND BEHAVIOR", Syracuse University..

[11] NGEL, J., et al., 1995. Consumer Behavior. International ed. ed. Florida: Dryden.

[12] Haubl, G., and Trifts, V." Consumer decision making in online shopping environments: the effects of interactive decision aids,.Marketing Science (19:1), 2000, pp. 4-21.

[13] Philip Kotler, Marketing Management Millenium Edition, Tenth Edition.

[14] Christy M. K. Cheung, Lei Zhu, Timothy Kwong,Gloria W.W. Chan, Moez Limayem, Online Consumer Behavior: A Review and Agenda for Future Research, 16th Bled eCommerce Conference,eTransformation Bled, Slovenia, 2003, University of Hong Kong, Information Systems Department.

[15] Churchill, G.A. and Peter, J.P. (1998). Marketing: Creating Value for Customers. Boston: Irwin/McGraw-Hill.

[16] Ayman-Nolley S. (1999), A Piagetian perspective on the dialective process of creativity. Creativity Research Journal, 12, 267–275.

[17] Guandong Xu،Yanchun Zhang،Lin Li," Web Mining and Social Networking: Techniques and Applications", Web Information Systems Engineering and Internet Technologies,Book Series, Victoria University, Australia.

[18] Mobasher B., Web Usage Mining and Personalization, in Practical Handbook of Internet Computing, CRC Press, 15, 1-37 (2004) .

[19] Atul Parvatiyar & Jagdish N. Sheth2, Customer Relationship Management:Emerging Practice, Process, and Discipline, Journal of Economic and Social Research 3(2) 2001, 2002 Preliminary Issue, 1-34.

[20] David Sogn,How Recommendation Engines Work Seeking Patterns in Reams of Data,By Advertising Age. Published on October 08, 2013.

[21] Myra Spiliopoulou"Web Usage Mining for web site evaluation",Communication of the web site evaluation of the ACM August 2000/Vol. 43, No. 8.

[22] Prachitee B. Shekhawat Prof. Sheetal S. Dhande,"A Classification Technique using Associative Classification", Dept. of Computer science and Engineering Sipna's College of Engineering and Technology, Amravati, Maharashtra, India, International Journal of Computer Applications (0975 – 8887),Volume 20– No.5, April 2011.

[23] Gerald Stermsek, Mark Strembeck, Gustaf Neumann," A User Profile Derivation Approach based on Log-File Analysis", Institute of Information Systems, New Media Lab Vienna University of Economics and BA, Austria.

[24] Dimitrio,Gergio Spaliours,Christos Patheodorou and Constatined Spyropoulos,"Web Usage Mining as a Tool for Personalization a survey", Institute of informatics and Telecommunications Department of Archive and Library Sciences, Ionian University, Corfu, GRGreece.

[25] Professor Dr. S.K. Jena, Karan Bhalla , Deepak Prasad," DATA PREPERATION AND PATTERN DISCOVERY FOR WEB USAGE MINING", Department of Computer Science & Engineering National Institute of Technology Rourkela 2007.

[26] Zdravko Markov and Daniel T .Larose ,Data Mining the Web:Uncovering Patterns in Web Content, Structure, and Usage, 2007 John Wiley & Sons.

[27] Gourab Kundu,Md. Faizul Bari, Sirajum Munir,"New Algorithms for Associative Classification", Department of Computer Science and Engineering Bangladesh university of Engineering and Technology June 2007.

[28] Liu B., Hsu W, and Ma W, "Integrating classification and associative rule mining", In KDD'98, New York, NY, Aug. 1998.

[29] Huiping Peng, "Discovery of Interesting Association Rules Based On Web Usage Mining", IEEE Coference, pp.272-275, 2010.

[30] Xindong Wu,Vipin Kumar,"The top ten algorithms in data mining ", 2009 by Taylor & Francis Group LLC

[31] Mustafa Nofal and Sulieman Bani-Ahmad," CLASSIFICATION BASED ON ASSOCIATION-RULE MINING TECHNIQUES: A GENERAL SURVEYAND EMPIRICAL COMPARATIVE EVALUATION",Department of Information Technology Al-Balqa Applied University Jordan, Al-Salt.

[32] Senthil K. Palanisamy," Association Rule Based Classification", WORCESTER POLYTECHNIC INSTITUTE ,in partial fulfillment of the requirements for the Degree of Master of Science In Computer Science May 2006.