Data Mining for Retail Website Design and Enhanced Marketing

Inaugural-Dissertation

zur Erlangung des Doktorgrades der
Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Asem Omari
aus Irbid

Juni 2008

Aus dem Institut für Informatik
der Heinrich-Heine Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Referent: Prof. Dr. Stefan Conrad

Koreferent: Prof. Dr. Martin Mauve

Tag der mündlichen Prüfung: 10.07.2008

{Allah will raise up, to suitable ranks and degrees, those of you who believe and who have been granted Knowledge. And Allah is well-acquainted with all ye do}

Translated from the Holy Quran 58:11

I would like to dedicate this thesis to my loving parents.

Acknowledgements

Every good comes through ALLAH alone. So all praises be to HIM.

I would love to express my appreciation to my supervisor Prof. Dr. Stefan Conrad, for investing plenty of time and effort to make my dissertation a success. Throughout my doctoral work he encouraged me to develop my scientific writing and research skills. I would like to thank Prof. Dr. Martin Mauve for reviewing my dissertation. I would like to thank my brother Dr. Tariq Omari and my friend Dr. Natheer Khasawneh for their invaluable comments while writing this dissertation.

There certainly exist no words that could possibly express the extent of gratitude I owe my loving mother and father and my caring and supportive brothers and sisters: Majdoleen, Osama, Sufian, Nuha, Monther, Tariq, and Omaia. Osama's support, especially during my stay in Germany, contributed to this achievement significantly.

I would like also to thank all my co-authors and undergraduate students who participated in the success of my dissertation. I thank my colleagues at the database and information systems group for creating such a nice research atmosphere. Very special thanks go also to Marga and Guido for not hesitating solving any management or technical problem I ever faced.

Abstract

Data mining is considered as one of the most powerful technologies that participates greatly in helping companies in any industry to focus on the most important information in their data warehouses. Data mining explores and analyzes detailed companies transactions. It implies digging through a huge amount of data to discover previously unknown interesting patterns and relationships contained within the company data warehouses to allow decision makers to make knowledge-based decisions and predict future trends and behaviors. Industries such as banking, insurance, medicine, and retailing commonly use data mining to reduce costs, enhance functionality, and increase sales. Web mining is the process of using data mining techniques to mine for interesting patterns in the web. Those patterns are used to study user behavior and interests, facilitate support and services introduced to the website navigator, improve the structure of the website, and facilitate personalization and adaptive websites.

In this dissertation, we developed a new approach that measures the effectiveness of data mining in helping retail websites designers to improve the structure of their websites during the design phase. This is achieved by giving them valuable information about the retail's

information system, its elements, and the relationships between different attributes of the information system. When considering this information in the design phase of the retail websites, they will have a positive effect in improving the website design structure. Furthermore, this approach reduces maintenance efforts needed in the future. We also studied the behavior of items with respect to time. This approach is beneficial in Market Basket Analysis for both physical and online shops to study customers buying habits and product buying behavior with respect to different time periods. We showed how association rule mining can be invested as a data mining task to support marketers to improve the process of decision making in a retail business. This is done through exploring current and previous product buying behavior and predicting and controlling future trends and behaviors. Based on our idea that interesting frequent itemsets are mainly covered by many recent transactions, a new method to mine for interesting frequent itemsets is also introduced. Finally, to solve the problem of the lack of temporal datasets to run or test different association rule mining algorithms, we introduced the *TARtool*. The *TARtool* is a temporal dataset generator and an association rule miner.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Contributions 5

1.3 Dissertation Organization 7

2 Knowledge Discovery **9**

2.1 The Knowledge Discovery Process 10

2.2 Data Mining 11

2.3 What Kind of Data Can be Mined 12

2.4 Data Mining Methods 15

2.4.1 Neural Networks 16

2.4.2 Case Based Reasoning 17

2.4.3 Decision Trees 17

2.4.4 Rule Induction 18

2.4.5 Data Visualization 18

2.5 Data Mining Tasks 18

2.5.1 Data Characterization 19

2.5.2 Clustering 19

2.5.3 Classification 21

2.5.4 Association Rule Mining 22
2.5.5 Sequential Pattern Mining 25
2.6 Data Mining Applications 26

3 Website Engineering 29

3.1 Software Engineering 29
3.2 Software Engineering Process 30
3.3 Web Engineering 32
3.4 E-Commerce and Retail websites 35

4 Data Mining in the Website Maintenance Phase (Related Work) 38

4.1 Web Usage Mining 39
4.2 Web Log File 40
4.3 Web Usage Mining Techniques 41
4.3.1 Statistical Analysis 41
4.3.2 Clustering 41
4.3.3 Classification 42
4.3.4 Association Rule Mining 42
4.3.5 Sequential Pattern Mining 43
4.4 Data Preprocessing for Web Usage Mining 43
4.4.1 Data Cleaning 43
4.4.2 Path Completion 44
4.4.3 User Identification 44
4.4.4 Session Identification 45
4.4.5 Session Formatting 45
4.5 Web Usage Mining for Adaptive Websites 45

4.5.1 Improving Website Usability and Organization 46
4.5.2 Adaptive Content 47
4.5.3 Adaptive Link 47
4.5.4 Adaptive Web Structure 48
4.5.5 Adaptive E-Commerce 48
4.6 Web Usage Mining for Personalized Websites 49
4.7 Web Content Mining 52
4.8 Web Structure Mining 53
4.9 Discussion 54

5 Data Mining in the Website Design Phase 57

5.1 Association Rule Mining During the Design Phase 59
5.1.1 Experimental Work 62
5.1.2 Method Evaluation 66
5.2 Classification and Clustering During the Design Phase 68
5.2.1 Experimental work 70
5.2.2 Method Evaluation 75
5.3 Datasets Availability 78

6 Temporal Frequent Itemset Mining for Enhanced Marketing 80

6.1 Related Work 80
6.2 Periodical Association Rule Mining 81
6.3 Temporal Frequent Itemset Mining 85
6.4 Experimental Work 90
6.5 Application Fields 93

7 Synthetic Temporal Dataset Generation **95**

7.1 Temporal Dataset Generation 96

7.2 Datasets for Association Rule Mining 97

7.3 Real World Versus Synthetic Datasets 98

7.4 Dataset Generators and Software Solutions 99

7.4.1 The IBM Generator 100

7.4.2 The DatGen Generator 100

7.4.3 An E-Commerce Generator 100

7.4.4 ARMiner 101

7.4.5 The ARtool Generator 101

7.4.6 WEKA 103

7.5 Enhancements for ARtool 103

7.6 Time Stamp Generation 105

7.7 Evaluation 109

8 Conclusion and Future Work **114**

8.1 Summary 114

8.2 Future Work 116

References **117**

List of Figures

3.1 Approximate Relative Costs of the Phases of the Software Process 32

5.1 Improved Website Design Structure Using Extracted Association Rules 60
5.2 Initial Website Prototype of the Grocery Store 63
5.3 Website Prototype With the Help of Extracted AR's 64
5.4 The Average Costs of Both Prototypes 68
5.5 Standard Website Design 71
5.6 A Summary of the Interesting Extracted Patterns 73
5.7 Improved Website Design 74
5.8 Average Session Times Needed to Simulate Customers Transactions in Both Websites 77
5.9 Time Needed to Finish 50 Transactions Manually 78

6.1 The Interestingness of Association Rules of a Set of Products With Respect to Different Time Periods 83
6.2 Interesting and Non-interesting Frequent Itemsets 88
6.3 An Example of Using the Necessary Condition 89

7.1 The ARtool GUI 102

7.2 The TARtool GUI 104
7.3 An Example of Generated Association Rules 107
7.4 Opening of an *.arff file in WEKA 110
7.5 A Comparison of Frequent Itemsets in TARtool and WEKA for the Same Generated Dataset 111

List of Tables

5.1 A Comparison of Session Times Within Different Clusters 77

7.1 Generation Costs for Binary and ASCII Files 112

Chapter 1

Introduction

1.1 Motivation

Recently, the web is becoming an important part of people's life. The web allows people to easily communicate and exchange ideas and views about any subject anywhere in the world. Furthermore, the web is a very good place to run successful businesses. Almost everything can be bought from online stores without the need to go to physical shops. Selling products or services online plays an important role in the success of businesses that have a physical presence, like a retail business. For many businesses, a retail website is an effective line of communication between the businesses and their customers. Even if the business does not present all of its products and services in the website, the website may be just what the customer needs to see to choose it over a competitor. Therefore, it is important to have a successful website to serve as a sales and marketing tool to participate in meeting the core requirements of the business.

A successful website is a well-structured website. The website is well-structured from the user's point of view if it contains services that satisfy user's needs, if the user navigation is simplified, and if he can reach his target page in a short time

without the need to make any search or to guess where his target page could be found. On the other hand, from the point of view of the website owner, a website is well-structured if it participates in increasing the business overall profit, if it increases the user's trust in the business and its products, and participates in supporting the business marketing strategies. Therefore, it is important to develop and use tools that can guarantee to a high degree the quality of websites to meet the requirements of both website owners and users.

One of the effective used technologies for that purpose is data mining. Data mining is the process of extracting interesting patterns from large databases. Web mining is the usage of data mining techniques to extract interesting information from web data. Patterns extracted from applying web mining techniques on web data can be used to maintain websites by improving their usability through simplifying user navigation and information accessibility and improving the content and the structure of the website in a way that meets the requirements of both website owner and user which will consequently increase the overall profit of the business or the industry that the maintained website belongs to [1].

Despite the effectiveness of web mining in improving websites, it costs a lot of maintenance efforts and suffers from different drawbacks. In commercial companies that sell different kinds of products online, in order to make an effective maintenance to their websites, the companies have to wait some period of time, for example one year, in order to have a representative log file that reflects customers transactions in their websites and can give a clear image about their behavior. This amount of time is considered very big especially for the companies in which the time factor plays an important role in their success strategy, and which have many competitors who can attract their customers if they have

no solid marketing strategies in order to keep their customers as loyal as possible. On the other hand, most businesses gather information about internet customers through online questionnaires. But, many customers choose not to complete these questionnaires because of the amount of time required to complete them as well as a lack of a clear motivation to complete them [2]. Several businesses use cookies to follow customers through the World Wide Web, but cookies are sometimes detected and disabled by web browsers and do not provide much insight into customer preferences. This is because customers are feeling that their profiles are not secure so a number of customers choose to give incorrect information about themselfs.

To overcome these problems, we made some steps back from the website maintenance phase to the design phase. In our approach (see [3; 4; 5; 6; 7]), the problem of building an ill-structured website for some retail business can be solved by applying data mining techniques such as association rule mining, clustering, and classification on the contents of the information system of the business. Then, from the extracted patterns, the information needed to be considered in the website building process is gained and invested during the design phase in the process of website design which yields to a better designed website. The main advantage of this method is the reduced maintenance time and budgetary costs for websites if they are built taking into account the extracted interesting patterns from the transactions database of the business. Furthermore, in web mining, different approaches are used to identify customers and transactions. Those approaches can not guarantee that the actual customers and transactions have been identified. Therefore, there exists a failure probability in defining customers and transactions. In contrast, in our methodology, we can guarantee that we mine the actual

customers transactions and profiles which were collected from users personally. For example, in a telecommunication company, when a customer wants to sign a mobile telephone contract, he usually fills a form that represents his profile. Any further products or services requested by the user will also be recorded. This approach also permits the sales manager to focus on the core business and gives him a better view about his products and customers which is very helpful in designing retail websites.

Another application field of data mining is using association rule mining to analyze market basket data. A transactions database contains information about customers transactions, where each transaction is a collection of items. Association rule mining captures the relationships between different items. An association rule finds the probability that two different items occur together in the transactions database. Association rule mining is finding all association rules that have support and confidence values greater than or equal a user-specified minimum support *(minsup)* and minimum confidence *(minconf)* respectively. *minsup* and *minconf* are functions that measure the interestingness of an association rule. Those rules are called interesting association rules. But the interestingness of an association rule that represents a group of items can have many different meanings. For example, an interesting rule may give some information about well-sold products. On the other hand, if we have a number of non-interesting association rules, we can also use them to gain some information about bad-sold products which is also considered a valuable information that can be invested by the marketers to improve their marketing strategies. In this approach, beside the usage of interesting association rules, the association rules that do not satisfy minimum requirements (i.e. have support and confidence values less than the user

specified *minsup* and *minconf*, respectively) are also considered in the decision making process [8]. Doing that in a periodical manner can be very effective in the decision making process.

Beside studying the behavior of frequent itemsets with respect to time, a new method for finding interesting frequent itemsets is also introduced. The core idea of this method is that the interesting frequent itemsets are mainly covered by many recent transactions [9]. The lack of suitable real life and synthetic datasets to apply our ideas motivated us to implement the *TARtool* which is a data mining tool and a dataset generator. The *TARtool* can generate temporal datasets that simulates both retail and e-commerce environments [10].

1.2 Contributions

This section describes the contributions of this work to the application of data mining in retail website design and in improving marketing strategies and targeting knowledge based decisions. The contributions are:

1. Developing a new method that measures the effectiveness of data mining in helping retail website designers to improve the structure of their websites during the design phase. This is achieved by investing interesting patterns extracted by applying different data mining tasks on the retail's information system during the design phase to support designing well-structured retail websites. The extracted patterns give the retail websites designers valuable information about the retail's information system, its elements, and the relationships between different attributes of the information system. Taking this information into account during the design phase of the websites

will reduce maintenance efforts needed in the future. It also overcomes the drawbacks of using web mining to maintain such websites. Furthermore, it makes it easy for the retail decision maker to design his retail website in a way that meets the main requirements and marketing strategies of his business which will consequently increase the overall profit of the business [3; 4; 5; 6; 7].

In this approach, we introduced two methods to evaluate the efficiency of websites. Time-based method evaluates the websites efficiency with respect to the time needed to finish a specific session. Link-based method evaluates the efficiency of websites in regards to the count of links that may be followed in the process of searching for the target products to finish a specific session (Chapter 5).

2. Demonstrating how association rule mining can be invested as a data mining task to support marketers to improve the process of decision making through predicting and controlling future trends and behaviors in the process of mining for association rules with respect to different time periods [8]. In this approach, in a retail industry, the information system of the retail and more specifically the customers transactions database is mined for association rules with respect to different time periods. The extracted patterns give information about current and previous buying behavior of products such as what are the well-sold and the bad-sold products. This gives the sales manager a better view about his products and their behavior, and help him to make right decisions and better marketing strategies. This also enables the sales manager to predict and control the sales behaviour in the next time

period which will consequently increase the retails overall profit (Chapter 6).

3. Developing a new method to search for interesting frequent itemsets in the process of association rule mining [9]. This method is based on the idea that the interesting frequent itemsets are mainly covered by many recent transactions. This method can be used either as a preprocessing step to search for frequent itemsets within a determined interval, or as an extension to the *Apriori* algorithm to prune non-interesting frequent itemsets (Chapter 6).

4. Developing the *TARtool* which is a temporal dataset generator and an association rule mining tool [10]. The *TARtool* can generate temporal datasets that simulates customers transactions in both retail and e-commerce environments. This tool solves the problem of the lack of temporal datasets to run and test different association rule mining algorithms for market basket analysis (Chapter 7).

1.3 Dissertation Organization

The rest of this dissertation is organized as follows. In Chapter 2, we present a general overview about knowledge discovery and data mining as an important phase of the knowledge discovery process. We discuss data mining techniques, applications, and tools. We also introduce web mining as an important application field of data mining.

In Chapter 3, we introduce a theoretical overview about website and software engineering due to their relativeness to our proposed method. We define them

and introduce their phases and advantages, and we talk about e-commerce and retail websites. In Chapter 4, we discuss in details the web mining process as a related work to our proposed methodology. Then, in Chapter 5, we introduce our methodology of investing data mining techniques during the design phase in designing retail websites. We start with a theoretical overview about the techniques and algorithms used. Then, we introduce our experimental work and the used methods to evaluate the results. We also discuss the problem of finding suitable datasets to run our experiments and introduce some application fields in the industry. In Chapter 6, we propose the concept of temporal data mining and frequent itemset mining and propose our contribution in investing temporal association rule mining in the decision making process especially in the field of marketing. We also introduce a new methodology to mine for interesting frequent itemsets. In Chapter 7, we discuss in details the problem of finding suitable datasets to run our experiments and test our proposed methodologies as one of the challenging problems that meet researchers especially in the field of temporal data mining. We introduce our developed tool, the *TARtool*, which is a temporal dataset generator and an association rule mining tool. Finally, in Chapter 8, we summarize the contributions introduced in this dissertation and present some highlights of the future work.

Chapter 2

Knowledge Discovery

The enormous expansion of the volume of collected data from different data sources and fields, yields to an increase of databases of all sizes and designs. However, it is obviously clear that the manual analysis of such amounts of data is impossible. For that reason, it is important to develop tools that can access and analyze such data to extract interesting knowledge that can be helpful in the decision making process. Knowledge discovery is a technique that can discover interesting information from such large databases. According to [11], "*Knowledge discovery in databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data*". The process is *nontrivial* means that it is not straightforward and some search methodology need to be followed. The extracted patterns should be *novel* or in other words unknown previously. They also have to be *useful*, this usefulness implies that the patterns can play a positive role in the decision making process. Finally, they must be *understandable* at least for the data analyst. The knowledge discovery in databases process consists of a number of iterative steps. A brief discussion of those steps is presented in the next section.

2.1 The Knowledge Discovery Process

The KDD process is interactive and iterative, involving a number of steps with many decisions made by the user [11]. Here we list the main steps of the knowledge discovery process:

- **Understanding the Application Domain**: The knowledge discovery process begins by understanding the application domain and the goals of the data mining process. Previously unknown patterns that are useful and effective in the decision making process are expected to be gained.

- **Data Integration and Gathering**: In the data integration and gathering step the target data sets are gathered from different data sources for example from heterogeneous databases and data warehouses and combined in a suitable manner. At this step, the relevant data to the analysis process is targeted and retrieved from the data source(s).

- **Data Preprocessing**: Eliminating errors, ensuring consistency, solving the problem of missed and repeated data, and transforming the selected data to format that are appropriate for the data mining procedure, are the main tasks of this step.

- **Data Mining**: Data mining is the most important step in the knowledge discovery process in which different techniques are applied to extract interesting patterns. In this step, the appropriate data mining algorithm(s) or/and method(s) are decided to be applied on the target data. A detailed discussion of data mining process is presented in the rest of this chapter.

- **Visualization**: To better understand and analyze the discovered knowledge, it is visually represented to the user using different visualization techniques.

- **Pattern Evaluation**: In this step, based on predefined measures, all interesting patterns representing meaningful knowledge are identified.

It is common to combine some of these steps together. For example, the data integration and gathering and the data preprocessing steps can be combined together and considered as one preprocessing step. Furthermore, most of those steps are often repeated iteratively to reach some satisfying level of the expected results. For example, in the data mining step the data analyst may repeat the data mining process by using another algorithm or method that could be more suitable for better and refined results of the knowledge discovery process. The data analyst can also jump within different steps. For example, if the data analyst is using some data mining algorithm and he want to use another algorithm that needs different data format, then he may go back to the data preprocessing step in order to convert the target data to the suitable format. In the next sections, we discuss data mining as the most important step in the knowledge discovery process.

2.2 Data Mining

Data mining is considered the most important step in the knowledge discovery process. Data mining is the process of extracting interesting patterns from large amounts of data [12]. It provides tools for automated learning from historical data and developing models to predict future trends and behaviors. Data mining

has two main models. *Predictive data mining model* tries to predict unknown value or future trends or behaviors depending on other variables or historical values presented in the mined database. For example, an immunization dose against some virus may be given to a particular patient not because he is infected from that virus, but because his illness profile is very similar to another group of patients who were infected with the same virus. *Descriptive data mining model* tries to extract useful patterns that can describe the mined data and explore its properties. For example, in a grocery store database, it may be found that a large number of customers who buy product A, buy product B with. Data mining is widely used by different companies and organizations especially in the field of retail, finance, communication, science, and marketing . It enables these companies to gain information about sales behavior, customer satisfaction, and corporate profits. Using data mining, organizations can increase the profitability of their interactions with customers, detect fraud, and improve risk management. The patterns extracted using data mining help organizations make better and knowledge based decisions. In the following sections, we discuss what kinds of data can be mined, data mining methods and tasks, and some data mining applications.

2.3 What Kind of Data Can be Mined

Data mining is not restricted to one kind of data store or repository, rather it can be applied to different kinds of data stores and repositories but with different algorithms and techniques that may be suitable to be applied on one kind of data but not for other. Data mining can be applied to flat files, relational databases,

data warehouses, transaction databases, multimedia databases, spatial databases, time-series databases, and the world wide web (WWW). Here we discuss briefly all those data repositories and mention some data mining application on that kind of data.

- **Flat Files**: Flat files are the most used data source in data mining. It is normally a simple data file with text or binary format with a structure that can be recognized by the used data mining algorithm for example, a text file with a comma separated format. The data in such files can be transactions, time-series data, medical measurements, etc.

- **Relational Databases**: A relational database consists of a set of tables. Each table has a number of columns and rows, where columns represent attributes, and rows represent tuples (records). Each tuple in a table represents an object or a relationship between objects and identified by a set of attributes that represent a unique key. SQL is the most common query language for relational databases. SQL can help in retrieving and managing the data stored in a relational database. For that reason, running data mining algorithms using relational databases can be more efficient and easier than using flat files due to the ability to take advantage of the functionalities that are provided by SQL such as data selection and transformation.

- **Data Warehouses**: According to [13], a data warehouse is defined as: "*A data warehouse is a subject oriented, integrated, time variant, and non-volatile collection of data in support of management's decision making process*". A data warehouse is a repository of data collected from multiple data sources and stored under a unified schema at a single site. The data

in a data warehouse can be loaded, preprocessed, and integrated together. The organization of the data warehouse with respect to different subjects gives the option to easily analyze the data and facilitates decision making process. Data warehouses are usually modeled as a multi-dimensional data structure (data cubes), where each dimension represents a set of attributes in the schema, and each cell stores a value of some aggregate measure [14]. Because of their structure and the precomputed summarized data, data cubes are well suited for interactive querying and analysis of data at different conceptual levels, known as *Online Analytical Processing (OLAP)*. OLAP allows the navigation of data at different levels of abstraction, such as, drill-down, roll-up, slice, dice, and pivot. As the data in the data warehouse are cleaned and integrated, there is a very small need for the data to be further cleaned and preprocessed to be mined.

- **Transactional Databases**: A transaction database is a set of records representing transactions, each with an identifier (Usually the transaction ID) and a set of items. The transactions database may have additional information that further describe and details such as the transaction date. One typical data mining application on such data is the so-called *Market Basket Analysis* that uses mostly association rule mining to study the relationship between different items in a transaction database.

- **Multimedia Databases**: Multimedia databases store images, audio, and video. A multimedia object is high dimensional which makes data mining a challenging process.

- **Time-Series Databases**: Time-series databases contain time related data such as stock market data or the distribution of car accidents in some region with respect to time. Data mining can be applied on such kind of data to study the changing behavior of items with respect to time.

- **The World Wide Web**: The WWW is a heterogeneous and dynamic data repository. It includes a huge number of data types varying from text, audio, video, raw data, and application. The WWW is composed of three major parts. The web content, which represents the documents available on the web. The structure of the web which is represented by the hyperlinks and the relationships between different documents in the web. And finally, the usage of the web, which describes how the web documents and resources are being accessed and used. The application of data mining techniques on the WWW data is called web mining. Web mining has three major approaches: Web content mining, web structure mining, and web usage mining. A detailed description of those approaches are discussed in Chapter 4.

2.4 Data Mining Methods

There are different data mining methods used to perform different data mining tasks. Those methods vary from statistical methods, neural networks, decision trees, genetic algorithms, rule induction, case based reasoning, and data visualization. In this section, we examine neural networks and case based reasoning as two well-known data mining methods. Furthermore, we discuss decision trees,

rule induction, and data visualization as we use these methods later in our experimental work.

2.4.1 Neural Networks

Neural Networks [15; 16] are analytic techniques modeled after the processes of learning in the neurological functions of the human brain and capable of predicting new observations (on specific variables) from other observations (on the same or other variables). Neural networks have the ability to derive meaning from complicated or imprecise data and can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. Neural networks use a set of processing elements (or nodes) that simulates neurons in the human brain. Processing elements are interconnected in a network that can then identify patterns in data once it is exposed to the data, i.e the network learns from experience just as people do. Since neural networks are best at identifying patterns or trends in data, they are well suited for prediction or forecasting needs including sales forecasting, industrial process control, customer research, risk management, and target marketing. Neural networks learn by example and are considered a predictive data mining method. The main goal of our approach is to extract patterns that describe current and past trends and behaviors, rather than predicting future trends and behaviors. Patterns extracted using neural networks methods do not give any explanation or declaration how those patterns are extracted or what kinds of information were the basics of getting some particular results. Applying neural networks in our approach is not helpful because we get no descriptive information about the mined dataset. Such descriptive information is essential in our approach.

2.4.2 Case Based Reasoning

Case based reasoning [17] is a method that tries to solve a given problem using past solutions of a similar problem by searching the existing set of case bases and finding a similar one. If a similar case exists, its solution is applied to the new problem, and the problem is added to the case database for future reference. In our approach, case based reasoning is not a suitable method to use as we have no specific problem such that we need to compare the characteristics of this problem with past examples or cases to find a one that fits.

2.4.3 Decision Trees

A decision tree [16; 18] is essentially a flow chart of questions or data points that lead to a decision. These decisions generate rules for the classification of a dataset. Decision tree systems try to create optimized paths, ordering the questions such that a decision can be made in the least number of steps. An example of a decision tree method is *Classification and Regression Trees (CART). CART* provides a set of rules that can be applied to a new (unclassified) dataset to predict which records will have a given outcome. Decision trees are great for situations in which a visitor comes to a website with a particular need so he can be assigned to his target need easily. Decision trees are well suited for our approach as they help in dividing customers and products into different groups and categories with respect to different attributes which enables an easy analysis and characterization of the target data.

2.4.4 Rule Induction

Rule induction [16] is the extraction of useful *if-then* rules from data based on statistical significance. Rule induction defines the statistical correlation between the occurrence among certain items in a dataset. One of the main data mining tasks that uses rule induction method is the association rule mining by which associations between different elements of the target data can be extracted which is very helpful in our approach to understand customer buying behaviours and trends (will be discussed in section 2.5.4).

2.4.5 Data Visualization

Data visualization [19] makes it possible for the analyst to gain more clear understanding of the data and focus on certain patterns and trends in the target data. Data visualization in its own is not enough to analyze the data due to the large volume of data in a database but with the help of data mining it can help in data exploration and analysis. In our experimental work, we used data visualization to have a visual overview about the extracted patterns which eases data analysis and exploration which is very helpful in targeting the right decisions.

2.5 Data Mining Tasks

The choice of which data mining task to use depends on the application domain and the structure of the patterns that are expected to be extracted. In the following sections, we discusses the basic data mining tasks: Data characterization, clustering, classification, and association rule mining. Those data mining tasks have been used in our experimental work as we will see in Chapter 5.

2.5.1 Data Characterization

Data characterization is summarization of the general characteristics or features of a target class of data [12]. To have a successful data mining process, it is always a good strategy to understand the target data (i.e. the dataset to be mined). This can be achieved by summarizing the target data and present it in a high conceptual level by using aggregate functions such as *Sum* and *Average*, using the Online Analytical Processing ($OLAP$) technique to explore the data with respect to different aspects and conceptual levels, or using data visualization to explore the data and display distributions of values in the dataset.

2.5.2 Clustering

Clustering divides a dataset into different groups. The objects in a dataset are grouped or clustered based on the principle that objects in one cluster have high similarity to one another but are very dissimilar to objects in other clusters [20]. In clustering data objects have no class label. That means when we start clustering we do not know what the resulted clusters will be, or by which attribute the data will be clustered. For that reason, clustering is also called *unsupervised learning*. Before running any clustering algorithm, the data analyst removes any irrelevant meaningless attributes. Clustering has many different methods and techniques. In the following we discuss the main clustering methods.

- **Partitional Clustering Method**: In partitional clustering method the dataset is divided into non-overlapping clusters such that each data object belongs to exactly one cluster. Objects in one cluster are similar or close to each other, but are dissimilar or far away from objects in other clusters.

To achieve well partitioned clusters different methods are used. Two major and well-known algorithms are the *K-Means* algorithm and the *K-medoids* algorithm. In the *K-Means* algorithm, each cluster is represented by the mean value of the objects in the cluster. In the *K-medoids* algorithm, each cluster is represented by one of the objects located near the center of the cluster [12]. Another well-known partitional clustering algorithm is the *Nearest Neighbor Algorithm*. This algorithm merges objects iteratively into the existing clusters that are closest. A threshold is used to determine if objects will be added to existing clusters or if a new cluster is created [21]. For example, to cluster a set of items, the first item of the set is placed in a cluster by itself. Then, we look at the second item and based on a distance threshold, we decide if it should be added to the first cluster or placed in a new cluster. This process is repeated until every item is placed in the suitable cluster.

- **Hierarchical Clustering Method**: If we permit a cluster to have sub-clusters, then we obtain hierarchical clusters. Hierarchical clusters are sets of nested clusters that are organized as a tree, where each node in the tree represents a cluster and its sub-nodes represent the sub-clusters. The root of the tree represents the cluster that contains all the objects. There are two main approaches in hierarchical clustering the *agglomerative* approach and the *divisive* approach. The *agglomerative* approach, starts with each object as a single cluster, and then repeatedly merge the two closest clusters until all clusters are merged in one cluster or a termination condition holds. In contrast to *agglomerative* approach, the *divisive* approach starts with a

one big cluster and repeatedly splits each cluster into smaller clusters until reaching the point that each object is in one cluster, or until a termination condition holds. *CURE* and *BIRCH* are two well-known hierarchical clustering algorithms [22].

- **Density Based Clustering Method**: In density based clustering method a cluster is a dense region of objects that is surrounded by a region of low density. In the *DBSCAN* density based clustering algorithm [23], the points in the low density regions are classified as noise and omitted.

2.5.3 Classification

Classification also known as *supervised learning* is the process of finding a set of models or functions that describe and distinguish data classes or concepts where the models derived based on a set of training data (i.e. data objects whose class label is known) [12]. The following are two well-known classification methods:

- **Decision Tree Based Classification Methods**: The decision tree classification method [24] is one of the most useful methods in data mining. A decision tree is a way of representing a series of rules that lead to a class or a value. The decision tree nodes are test values that need to be decided or questions that need to be answered in order to decide the branches that should be followed depending on the decided value. Each node may have two or more branches. Each branch will lead to another decision node or to the bottom of the tree, which is called the leaf node. To assign a value to some class or case, the navigation process starts at the root node and moves to each subsequent node and decide which branch to take until reaching the

leaf node. Decision trees algorithms such as *C4.5* [25] and *CART* [26] are used in classification to make predictions of current and future values of some attributes.

- **Distance Based Classification Methods**: One of the simplest algorithms that is based on distance based classification method is the *k-Nearest Neighbor* algorithm [21]. It classifies an object based on the class of its nearest neighbor in the training set. A straightforward generalization of this approach is to classify an object as belonging to the class which is most frequently represented among its k nearest neighbors. The choice of k is data dependent. Usually, k depends on the size of the training set.

2.5.4 Association Rule Mining

An association rule is an expression of the form $X \Rightarrow Y$, where X and Y are sets of items and have no items in common. This rule means that given a database of transactions D where each transaction $T \in D$ is a set of items. $X \Rightarrow Y$ denotes that whenever a transaction T contains X then there is a probability that it contains Y too. The rule $X \Rightarrow Y$ holds in the transactions set T with confidence c if $c\%$ of transactions in T that contain X also contain Y. The rule has support s in T if $s\%$ of the transactions in T contains both X and Y. Association rule mining is finding all association rules that are greater than or equal a user-specified minimum support ($minsup$), and minimum confidence ($minconf$). In general, the process of extracting interesting association rules consists of two major steps. The first step is finding all itemsets that satisfy $minsup$ (known as *Frequent-Itemset* generation). The second step is generating all association rules

that satisfy $minconf$ using itemsets generated in the first step.

According to [27], in the process of searching for frequent itemsets, Association rule mining algorithms employ one of two common approaches: Breadth-first search approach (BFS), and Depth-first search approach (DFS). In BFS approach, the support values of all $(k-1)$-itemsets are determined before counting the support values of the k-itemsets where k is a positive integer. Supposing that the transactions data are represented in a tree structure, in DFS approach the algorithm can start from, say, node a in the tree and counts its support to determine whether it is frequent. If so, the algorithm expands the next level of nodes until an infrequent node is reached. It then backtracks to another branch and continues the search from there.

The Apriori algorithm [28] follows the Breadth-first search approach. It generates all frequent itemsets, called also large itemsets, by making multiple passes over the transactions database D. The algorithm makes a single pass over the data to determine the support of each item which results in the set of 1-itemsets. Next, the algorithm will iteratively generate new candidate k-itemsets using the frequent $(k-1)$-itemsets found in the previous iteration. An Additional pass over the dataset is made to count the support of the candidates. After counting their supports, the algorithm eliminates all candidate itemsets whose support count are less than *minsup*. The algorithm eliminates some of the candidate k-itemsets using the support-based pruning strategy. If any subset of the k-itemset X is not frequent, then X is pruned. The algorithm terminates when there is no new frequent itemsets generated. Association rules are generated by generating all non-empty subsets of each frequent itemset and outputs its rule if its confidence is greater than or equal *minconf*.

The *AprioriTID* algorithm differs from the *Apriori* algorithm in that it does not use the database D for counting support after the first pass. Rather, it uses the set of candidate k-itemsets associated with the transactions identifiers (TID's), so that the number of entries in this set may be smaller than the number of transactions in the database, especially for large values of k [28]. *The AprioriHyprid* Algorithm is an algorithm that get benefit from both *Apriori* and *AprioriTID* algorithms in which it starts by using the *Apriori* algorithm, then it switches to *AprioriTID* in the last passes. But if there is no candidate itemsets found in this stage then we just pay the cost of switching to *AprioriTID* without getting benefit of using it [28].

Unlike the *Apriori* algorithm, the *FP-Growth* algorithm follows the Depth-first search approach. *FP-Growth* mines frequent itemsets without candidate generation. It encodes the dataset using a compact data structure called the *FP-Tree*, and extracts frequent itemsets directly from this structure. The *FP-Tree* [29] is created as follows: First, a root node of the tree is created and labeled "null". For each transaction in the database, the items are processed in reverse order and a branch is created for each transaction. Every node in the *FP-tree* stores a counter which keeps track of the number of transactions that share that node. When adding a branch for each transaction, the count of each node among the common prefix is incremented by 1, and nodes for the items in the transaction following the prefix are created and linked accordingly. Additionally, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links. Each item in this header table also stores its support. The transactions in the *FP-Tree* are stored in support descending order which keeps the *FP-tree* representation of the database as small as possible since the more

frequently occurring items are arranged closer to the root of the *FP-tree* and thus are more likely to be shared. Because there is often a lot of sharing of frequent items among transactions, the size of the tree is usually much smaller than its original database which avoids the costly database scans implemented by *Apriori*-like algorithms.

The study in [30] shows that the *FP-Growth* has several advantages than any other *Apriori*-like algorithms, especially when the dataset contains many patterns and/or when the frequent patterns are long. In an experimental application on different real-world and artificial datasets in [31], a comparison between *Apriori, FP-Growth* and other algorithms shows that the *Apriori* outperforms the *FP-Growth* when the *minsup* value is small. But with high *minsup* values, the *FP-Growth* outperforms the *Apriori*. Another experiments on real-world and artificial datasets are presented in [32]. The experiments show that the *FP-Growth* performs best in comparison with *Apriori* and other implemented algorithms.

2.5.5 Sequential Pattern Mining

In sequential pattern mining a sequence of actions or events is determined with respect to time or other sequences [33]. The problem of mining sequential patterns over transactional databases is introduced in [34]. The authors presented three algorithms and evaluated their performance using synthetic datasets. *SPAM* is an algorithm for finding all frequent sequences within a transactional database. The algorithm is especially efficient when the sequential patterns in the database are very long. A depth-first search strategy is used to generate candidate sequences, and various pruning mechanisms are implemented to reduce the search space [35].

Further details and discussions concerning different clustering, classification, association rule mining, sequential pattern mining, and other data mining methods and algorithms can be found in [12; 20; 21].

2.6 Data Mining Applications

Data Mining is becoming increasingly popular because of its wide applicability in many different fields. Data mining made substantial contributions to the success of different industries and application fields. In the following, we examine the application of data mining in science, business and other application domains.

- **Data Mining in Customer Relationship Management**: Customer Relationship Management (CRM) tries to use all measures to understand customers and invest the extracted knowledge to implement marketing strategies, control production, and coordinate the supply chain. Data mining is one of those tools used by CRM to study customer behavior and interests and use the extracted knowledge to manage customers life cycle, attract new customers, and retain good customers. For example, by analyzing customer profiles, we can find the group of customers who bought a particular product and encourage similar customers have not bought that product to buy it. Furthermore, by studying the profiles of customers who have left the company, a special strategy can be built to retain customers who are expected to leave, because it is less expensive to retain an existing customer that to win a new one. The authors in [36] employ a data mining tool to discover current spending patterns in a credit card business of customers and

trends of the behavioral change to expand the customers base and prevent loosing of customers.

- **Data Mining in Marketing**: Market basket analysis uses data mining techniques to analyze customers transactions in a retail transaction database. The extracted knowledge can be used for cross-selling, store design, discount plans, and promotions. It is also useful for fast interaction with each new customer of the company. Although market basket analysis investigates shopping carts and supermarket shoppers, it is important to realize that there are many other areas in which it can be applied such as analysis of credit card purchases and telephone calling patterns.

- **Data Mining in Insurance**: Companies in the insurance industry collect enormous amounts of data about their clients such as information about customers behavior, activities, and preferences. Applying data mining on such data might be useful in detecting and predicting fraud or risky customers and predicting which customers will get benefit from new services and policies.

- **Data Mining in Science**: Data mining has many different applications in different fields of science [37]. In bioinformatics it can be used to analyze DNA sequences to find the relationship between some specific gene sequence and a specific disease. In other words, the change of the DNA sequence can give an indication of the high probability of the development of some disease. Data mining is used also in medical decision support. Patient records include patients demographic information, symptoms, blood measurements and laboratory test results. Mining those data can find the

correlation between two different diseases, or the relationship between social and environmental issues and some diseases, or how a group of patients react to a specific drug. In astronomy where huge number of images is collected from telescope and satellites, data mining can be used to classify objects such as stars and galaxies depending on objects attributes. It can also be used to find rare, unusual, or even previously unknown types of astronomical objects and phenomena.

- **Web Mining**: Applying data mining techniques to extract interesting pattern from web data is called web mining. Web mining is divided into three major categories, Web usage mining, web content mining and web structure mining. A detailed discussion of those techniques and their applications is presented in Chapter 4.

- **Data Mining in Telecommunications**: In telecommunication companies, huge amount of data is collected daily varying from transactional data to other customer data such as billing information or customers profiles, and additional data such as network load. Data mining can be used in the field of telecommunications [38] to find customer groups that are highly profitable, decide which services or products should be offered to which customers, and find which kind of call rates that can increase the profit without loosing good customers. Fraud detection techniques can also help in finding stolen mobile phones or phone cards.

In the next chapter we examine website engineering as a strongly related subject to our approach.

Chapter 3

Website Engineering

3.1 Software Engineering

The amount of time, effort, and money spent on software development and software use is huge. Software is involved in almost everything we do, business and banking, medical care, public transportation, government operations, human communication, education, etc. Companies in every business expend a lot of time and effort planning, designing, and creating software.

For that reason, it is important to follow a series of steps that ensures a timely and high-quality results when building a software. This is done using the so-called *software engineering*. The IEEE [39] defines software engineering as: "*Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software.*" Software engineering is the establishment and use of sound engineering principles in order to obtain software that is reliable and efficient. According to [40], the work associated with software engineering can be categorized into three generic phases. In the *definition phase*, the software engineer tries to identify what information is to be processed, what is the expected functionality and performance of the

engineered software, what design constraints and validation criteria are required. The *development phase* focuses on the way software design, code generation, and software testing are performed. The *support phase* involves changes associated with error correction, adaptations required to the software environment, and adaptations needed due to changes in customer behavior and requirements. In the next section, we discuss in detail different phases in the software engineering process.

3.2 Software Engineering Process

The core idea of the software engineering process is to divide the software building process into relevant phases that can be dealt with separately and iteratively. Every phase of the software engineering process focuses on a specific problem or a challenge taking into account the requirements of other phases. The process of walking through those phases are called the software life cycle. These phases can differ depending on the application field, but in general they can be summarized as follows:

- **Requirement Analysis and Specification Phase**: In this phase, the problem need to be solved is identified. The operational capabilities, the expected performance, the goals need to be achieved from building this software, and the characteristics of the software are also identified.

- **Design Phase**: In the software design phase, software requirements are analyzed in order to produce a description of the software's internal structure that will serve as the basis for its construction. More precisely, a software design must describe the software elementary prototype and architecture

that is, how software is decomposed and organized into components and the interfaces between those components.

- **Coding Phase**: In this phase, the requirements of the previous phases are implemented using a suitable tool or programming language.

- **Integration and Testing Phase**: Through verifying the consistency and completeness of the implemented models, integration and testing sustains the overall integrity of the software system architectural configuration [41]. One purpose of testing is to reduce the gap between the time at which errors are inserted into the code and the time those errors are detected. Furthermore, the functionality of the software is tested taking into account many factors, for example, taking into account that customers can use the software under different operating systems, testing the functionality of the software in different operating systems is required in this case.

- **Maintenance Phase**: Software maintenance sustains the software product through its life cycle. The maintenance of an existing software can account for over 60 percent of all effort expended by a development organization [40]. According to [40], *"there are four different maintenance activities: Corrective maintenance, perfective maintenance or enhancement, adaptive maintenance, and preventive maintenance or reengineering"*. According to the author, *"only about 20% of all maintenace work is spend on "fixing mistakes". The remaining 80% is spent adapting existing systems to changes in their external environments, making enhancements requested by users, preventing system performance from degrading to unacceptable levels, and reeingeneering an application for future use."*

Figure 3.1 (modified from [42]) shows the approximate relative costs of the phases of the software process. As seen in the figure, the maintenance phase consumes approximately 67% of the overall software process. On the other hand, the approximate cost of any other phase does not exceed 7% of the software engineering process. This highlights the need to use new technologies/tools that participate in reducing maintenance costs.

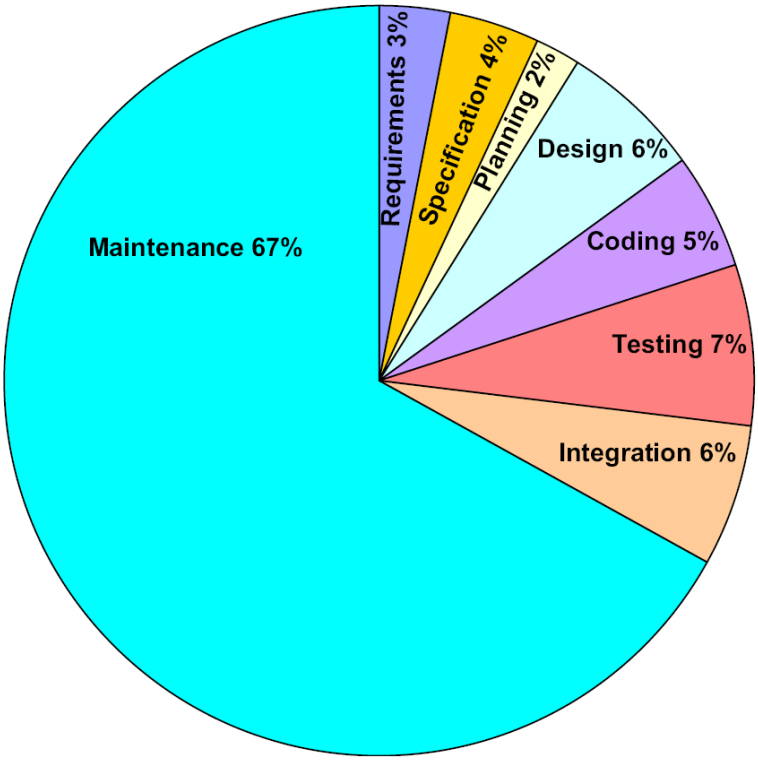


Figure 3.1: Approximate Relative Costs of the Phases of the Software Process

3.3 Web Engineering

In the past few years, many software engineers are interested in the way web application are being developed. As the web applications need to be reliable

and perform well, the core question is whether software engineering techniques and methodologies can be applied to the website development process. Software engineers found that many of software engineering principles, concepts and methods can be applied to web development, but there are subtle differences in the way these activities are performed [40; 43]. This leads to the principle of web engineering. The authors in [40] define web engineering as: "*Web Engineering is concerned with the establishment and use of sound scientific, engineering, and management principles and disciplined and systematic approaches to the successful development, deployment, and maintenance of high quality web based systems and applications*". Several attributes of quality web-based systems such as usability, navigation, accessibility, scalability, maintainability, compatibility and interoperability, security, and reliability are often not given the consideration they deserve during development [44].

Web development is not just creating web pages that seem beautiful to a user. While web engineering uses software engineering principles, it encompasses new approaches, methodologies, tools, and guidelines to meet the unique requirements of the web-based systems. In general, web development process consists of the following general phases:

- **Requirement Analysis Phase**: In this phase the general objectives and requirements of the developed website are identified, as well as the requirements of the website's users. The information need to be on the website, how to get this information, and how often this information may change. In [44], the authors recommended several key steps that need to be considered for successful development and deployment of web applications such

as understanding the system overall function and operational environment including business objectives and requirements and identifying the system main users and their typical profiles. According to [45], The secret of a successful web-based business is to define the characteristics of potential customer groups and their interest in specific set of products/services. This facilitates planning an effective marketing strategy.

- **Design Phase**: In this phase, different components of the website are decided and the links between them are defined. In the design phase, it is a good practice to try to guarantee the consistency of information that appear in different places and pages. Web page content development needs to take into consideration the website owner's requirements, user's abilities, technical issues and considerations, earlier experiences of developers and users, and lessons learned from similar web applications [44]. A lot of research has been done to cover different website design techniques, and strategies. The work in [46] provides a survey of experts' recommendations of how to create an effective website from an e-commerce point of view. It investigates the determinants of an effective website. The survey indicated that the major categories of determinants are: page loading speed, business content, navigation efficiency, security, and marketing/customer focus. The authors in [47] and [48] present a method for designing kiosk websites which are websites that provide information and allow users to navigate through those information. The method is based on the principle that the website should be designed and adapted to its users. It starts by identifying different classes of users and describe manually their characteristics, and their

information requirements, and how could they navigate the website. The work in [49] gives some recommendations and remarks on how to design retail websites. For example, stores that offer a FAQs section have more visits than those without such a section and every web page must have consistent navigation links to move around on the website.

- **Testing Phase**: Before the website is being published, it is tested and the found errors are corrected. The website engineer makes sure that every component is correctly built and working correctly.

- **Maintenance Phase**: After the website is developed and published online for use, it need to be maintained, based on the decision taken at the design phase on how the information content would be maintained. As the requirements of the website users and owners change over time, the website needs to be updated and redesigned to include the new requirements. Website maintenance is not only a matter of updating HTML files, it is concerned with ensuring links availability and consistency, adapting the web pages to follow new technologies, improve web page structure and content to meet the requirements of both website owner and the user who navigates the website in a way that increases the benefit of the website owner by increasing sales and/or reducing costs, and makes the user satisfied navigating the website and getting benefit from the services introduced to him.

3.4 E-Commerce and Retail websites

In e-commerce instead of having your business in a limited physical place and a limited sector of customers who are usually near to your store or business, you

have it in the web. In e-commerce websites you have the ability to sell, advertise, and introduce different kinds of services and products in the web. E-commerce websites have the advantage of reaching a large number of customers regardless of distance and time limitations. Furthermore, an advantage of e-commerce over traditional businesses is the faster speed and the lower expenses for both e-commerce website owners and customers in completing customers transactions and orders.

Because of the above advantages of e-commerce over traditional businesses, a lot of industries in different fields such as retailing, banking, medical services, transportation, communication, and education are establishing their business in the web. But creating a successful online business can be a very difficult and costly task if not taking into account e-commerce website design principles, web engineering techniques, and what e-commerce is supposed to do for the online business. Understanding the requirements of both e-commerce website owner and customer is an important aspect in building a successful e-commerce website. The work in [45] discusses a lot of key information need to be defined before starting building the e-commerce website such as identifying business goals and how the website will target those goals, if the website supposed to attract new customers or increase the sales of current customers, identify if the proposed website will increase the business overall profit, and identify the most suitable tools and techniques need to be used/followed in order to target those requirements. Retail websites aim to inspire, reflect a good image about the business and improve it online. An important factor in having a successful retail website is to know your competitors. On one hand, by identifying their points of strongness and trying to get benefit of them by improving those strongness points and adopting powerful

strategies. On the other hand, identifying weakness points of your competitors and avoid them is a good practice in having a successful retail website.

In the next chapter we discuss the problem of using web mining techniques in the maintenance phase of websites to improve the efficiency of retail websites.

Chapter 4

Data Mining in the Website Maintenance Phase (Related Work)

The usage of data mining to maintain websites and improve their functionality is an important field of study. In this chapter, we will discuss web mining and the available approaches used to maintain and adapt retail websites to meet the requirements of both the website owner and the customer navigating the website. In this chapter, we will investigate different web mining categories and more specifically web usage mining as a very related subject to our approach. We will discuss different data mining tasks that are mostly used in web usage mining. Then, we will talk about different data preprocessing approaches adopted to make the web log file ready to be mined. After that, we discuss web usage mining approaches to have adaptive and personalized retail websites. And finally, we summarize the advantages and disadvantages of all of the above approaches.

4.1 Web Usage Mining

Web mining is the use of data mining techniques to extract useful patterns from the web. Those extracted patterns are used to improve the structure of websites, improve the availability of the information in the websites and the way those pieces of information are introduced to the website user, and to improve data retrieval and the quality of automatic search of information resources available in the web. Web mining can be divided into three major categories: web usage mining, web content mining, and web structure mining.

Web usage mining or web log mining is the process of applying data mining techniques to web log data in order to extract useful information from user access patterns [1]. Web usage mining tries to make sense of the data generated by the web user's sessions or behaviors [50]. The web usage data includes data from web server access log, proxy server logs, browser logs, user profiles, registration data, cookies, and user queries [50]. Web usage mining tries to predict user behavior while user interacts with the web and learns user navigation patterns. The learned knowledge could then be used for different applications such as website personalization, business intelligence, usage characterization and adaptive websites. There are two approaches for web usage mining process [51]:

- Mapping the log data into relational tables before an adopted data mining techniques is performed.
- Using the log data directly by utilizing special preprocessing techniques.

The Web usage mining process consists of three phases: *data preprocessing, pattern discovery*, and *pattern analysis*. Pattern discovery is that set of methods,

algorithms, and techniques used to extract patterns from web log file. Several techniques are used for pattern discovery such as statistical analysis, clustering, classification, and sequential pattern mining (see section 4.3). After patterns are discovered they need to be analyzed in order to determine interesting and important patterns, besides the removal of redundant patterns. Pattern analysis has several different forms such as knowledge query mechanism, visualization techniques, and loading usage data into a data cube in order to perform *Online Analytical Processing OLAP* operations [52].

4.2 Web Log File

A web server log file records users transactions in the web. Usually, the web log file contains information about the user IP address, the requested page, time of request, the volume of the requested page, its referrer, and other useful information. The web log file can have different format, but there is a common log file format that is mostly used. The common log file has the following format [53]:

```
remotehost rfc931 authuser [date] "request" status bytes
```

remotehost represents remote hostname (or IP number if DNS hostname is not available), *rfc931* represents the remote logname of the user, *authuser* represents the username as which the user has authenticated himself, [*date*] represents date and time of the request, *"request"* represents the request line exactly as it came from the client, *status* represents the HTTP status code returned to the client, and finally *bytes* represents the content-length of the document transferred. The WWW Consortium (W3C) presented an extended format for web server log file [54] that is able to record a wide range of data to make an advanced analysis of

the web log file. Web log file is the main source of data analysis in web mining. A lot of preprocessing efforts need to be performed in order to prepare the web log file to be mined as we will see in the next sections.

4.3 Web Usage Mining Techniques

In this section, we discuss data mining techniques that are mostly used in web usage mining such as statistical analysis techniques, clustering, classification, association rule mining, and sequential pattern mining.

4.3.1 Statistical Analysis

Statistical analysis is the process of applying statistical techniques on web log file to describe sessions, and user navigation such as viewing the time and length of a navigational path [52]. Statistical prediction can also be used to predict when some page or document would be accessed from now [55]. The work in [51] makes use of the *N-grammer* model which assumes that when a user is browsing a given page, the last N pages browsed affect the probability of the next page to be visited.

4.3.2 Clustering

Clustering is the process of partitioning a given population of events or items into sets of similar elements [12]. In web usage mining there are two main interesting clusters to be discovered: usage clusters, and pages clusters [52]. The authors in [56] present an approach to cluster web pages to have a high quality clusters of web pages and use that clusters to produce index pages, where index pages are web pages that have direct links to pages that may be of interest of some

group of website navigators. In [57] clustering techniques are applied to web log file to discover those subsets of web pages that need to be connected, and to improve the already connected pages. The work in [58] uses the *Competitive Agglomeration Clustering Algorithm* to cluster the sessions extracted from web log server into typical session profiles of users. The authors in [33] use a clustering algorithm which identifies groups of similar sessions, allowing the analysis of visitor behavior.

4.3.3 Classification

Classification is dividing an existing set of events or transactions into another predefined sets or classes based on some characteristics. In web usage mining, classification is used to group users into predefined groups with respect to their navigation patterns in order to develop profiles of users belonging to a particular class or category [52]. [59] introduces several approaches for web page classification. The authors in [60] propose an approach to reorganize a website based on user access patterns and the classification of web pages into two categories: index pages and content pages.

4.3.4 Association Rule Mining

Association rule mining is the discovery of attribute values that occur frequently together in a given set of data [12]. Association rules mining techniques are used in web usage mining to find pages that are often viewed together, or to show which pages tend to be visited within the same user session [61]. The work introduced in [62] proposes a re-ranking method with the help of website taxonomy to mine for generalized association rules and abstract access patterns of different levels to

improve the performance of site search. The authors in [63] propose an approach for predicting web log accesses based on association rule mining. Association rule mining facilitates the identification of related pages or navigation patterns which can be used in web personalization [1; 64; 65].

4.3.5 Sequential Pattern Mining

In sequential pattern mining a sequence of actions or events is determined with respect to time or other sequences [33]. In web usage mining, sequential pattern mining could be used to predict user's future visit behaviors. Some web usage mining and analysis tools use sequential pattern mining to extract interesting patterns such as *SpeedTracer* [66] and *Webminer* [67]. The authors in [68] suggest using adaptive websites to attract customers using sequential patterns to display special offers dynamically to them.

4.4 Data Preprocessing for Web Usage Mining

Before data mining techniques are applied to web log file data, several preprocessing steps should be done in order to make web log file data ready to be mined. Web log file contains data about requested URL, time and date of request, method used, etc. The main data preprocessing tasks are data cleaning and filtering, path completion, user identification, session identification, and session formatting.

4.4.1 Data Cleaning

Data cleaning is the first preprocessing task. It involves the removal or elimination of irrelevant items that are not important for any type of web log analysis. Elimination of irrelevant items can be accomplished by checking the suffix of the

URL name to filter out requests for graphics, sound, and video hits in order to concentrate on data representing actual page hits [67; 69]. For example, all log entries with filename suffixes such as gif, jpeg, and jpg can be removed. Another cleaning process is removing log entries generated by web agents like web spiders, indexers, or link checkers [69]. Filtering out failed server requests, or transforming server error code is also done. Merging logs from multiple servers and parsing the log into data fields is also considered a data cleaning step [70].

4.4.2 Path Completion

Path completion preprocessing task fills in page references that are missing due to local browsing caching such as using the back button available in the browser to go back to previously visited page [71].

4.4.3 User Identification

Identifying unique users is a complex step due to the existence of local caches, corporate firewalls, and proxy servers [67]. If the agent log shows a change in browser software, or operating system, a reasonable assumption to make is that each different IP address in the log file represent a different user [72]. If a page is requested that is not directly reachable by a hyperlink from any of the pages visited by the user, a heuristic assume that there is another user with the same IP address. Another assumption can be made is that consecutive accesses from the same host during a certain time interval come from the same user [73]. In some cases it is difficult to identify users, for example, when two users use the same machine and the same browser with the same IP address and look at the same set of pages [71].

4.4.4 Session Identification

A user session is defined as *"the set of pages visited by the same user within the duration of one particular visit to a website"* [72]. Session identification is dividing the page accesses of each user into individual sessions. One approach to identify user sessions, is by using a timeout threshold that is if the time between pages requests exceeds a certain limit (e.g. 30 minutes), then the user is starting a new session [71; 74]. Another approach assumes that consecutive accesses within the same time period belong to the same session [73].

4.4.5 Session Formatting

A final preprocessing step could be formatting the sessions or transactions for the type of the data mining technique, or algorithm to be applied [71]. The *Webminer* in [67] formats the cleaned web server log data in order to apply either association rule mining or sequential pattern mining.

4.5 Web Usage Mining for Adaptive Websites

Adaptive websites are *"websites that semi-automatically improve their organization and presentation by learning from user access patterns"* [75]. A site ability to adapt should be enhanced with information about its content, structure, and organization. For example, to add a link to a list of links ordered alphabetically, the link should be added at a specific point in the list. In the following subsections, we categorize different approaches of adaptive websites, even though it is difficult to make borders between different adaptation approaches for example, improving website links yields consequently to improve the structure of the website.

4.5.1 Improving Website Usability and Organization

Improving website usability can be achieved through making changes to the organization of the pages and links of the website. The work in [60] aims to build an adaptive website that will reorganize its pages so that its users can find the information they want with minimum effort, where *effort* is defined in [75] as *"a function of the number of links traversed and the difficulty of finding that links in website pages"*. Reorganization process is done by firstly extracting access patterns from web server's log file. Secondly, the web pages in the web sever are classified into index pages and content pages based on the characteristics and access statistics of the pages. Finally, the whole website is analyzed and a reorganization of the website is presented based on access information and page classification.

The authors in [76] propose an algorithm to automatically find pages in a website whose location is different from where visitors expect to find them. The expected locations are then presented to the website administrator to add a navigation link from the expected location to the target page. The authors also present another algorithm to select the set of navigation links to optimize the benefit to the website or the visitor. The approach discussed in [77] presents a model to improve the success of the website with the help of data mining techniques. To evaluate the efficiency values of a site pages, the authors analyze the navigational behavior of the site visitors with web usage mining. The analyst may decide to perform navigation pattern discovery over the entire log or to split it into customer log, or non-customer log and performs a comparative analysis of the two. Then makes decisions depending on the discovered results. In [78] a framework that enables

adaptation of the web topology and ontology to the needs and interests of web users is introduced. The proposed adaptation process exploits the access data of the users together with the semantic aspect of the web in order to facilitate web browsing.

4.5.2 Adaptive Content

Changing the content of a website can make the website better serve the requirements of a specific user. Content may be added, removed, or rearranged [79]. This includes additional explanations or details which may be added or removed depending on user's background and interests in some topic, or changing the website presentation language based on the user language preference.

4.5.3 Adaptive Link

Making changes to the links of the website can facilitate user's navigation of the website and minimize the time required to reach the target page. There are several techniques for adaptive link such as direct guidance and link sorting, hiding, disabling, or highlighting. Direct guidance technique provides the user with a link to the page which is predicted to be the best next step for the user [80]. The AVANTI project [81] tries to predict user's goals and presents links leading directly to pages it thinks a user will want to see. The work in [82] proposes an approach to suggest a path to unexperienced users if many users follow the same path in their search for information. Link sorting is done by selecting the most relevant pages based on the users interests or goals then sorting them based on their relevance and presenting them in an ordered list of links [79; 80]. Hiding or disabling the links that are not relevant to the user interests and goals makes

the user less confused and speeds up user's navigation [79; 83]. Link highlighting can also facilitate user's navigation [80; 83].

4.5.4 Adaptive Web Structure

Adding or removing new pages is a final decision of the website administrator. Depending on the extracted usage patterns, several changes may be done on website structure. The authors in [75] investigate the creation of index pages, which are pages that contain a direct link to pages that cover a particular topic, to facilitate the user's navigation of the website. The *PageGather* cluster mining algorithm is introduced. It takes web server logs as input and finds collections (clusters) of pages that tend to co-occur in visits, and outputs the contents of candidate index pages for each cluster found. A further development to [75] is found in [84] by presenting the *IndexFinder* a conceptual clustering mining algorithm in which all discovered clusters have intuitive descriptions that can be expressed to human users to solve the problem that *PageGather* gives no guarantee that all objects in the discovered cluster are about the same topic. To measure the use of a set of pages [79], statistics about commonly viewed pages and subsets of pages is generated. The administrator can get an idea how the structure of the web should be, and whether there are some pages need to be removed, added, or their position need to be changed, without destroying the overall structure of the website.

4.5.5 Adaptive E-Commerce

Web usage mining has a great effect on e-commerce. It can be used to study customer behavior in the web, and use the extracted knowledge to facilitate nav-

igation and services introduced to the customer, and suggest some particular products to the customer based on his interests. In [85] comparisons of navigation patterns between customers and non-customers lead to rules that specify how the website should be improved. The work in [68] suggests using adaptive websites to attract customers using sequential patterns to display special offers dynamically to them, and to keep the online shopper as loyal as possible. An example of e-commerce site that uses personalization is amazon.com, in which recommendations are presented to different customers depending on the customer profile [86].

In order to make websites more effective to website users, they should reflect their interests, knowledge, needs, and goals. This can be done through personalization which is the subject of the next section.

4.6 Web Usage Mining for Personalized Websites

Web personalization is the process of customizing websites to the needs of specific users taking advantage from the patterns discovered from mining web usage data and other information such as web structure, web content, and user profile data [73]. Web personalization begins with the collection of web data. In this stage usage data are collected from different sources such as web server side data, client side data, and proxy servers.

In general, personalization techniques are divided into offline and online techniques. Offline personalization is based on simple user profiling and manual decision rule systems. Web usage mining is an online personalization data source.

By evaluating site behavior and usage, a view about the website user is gained which yields to more effective personalization strategies. User profiles are an important source of data for data personalization. User profiles contain user preferences, characteristics, interests knowledge, skills, activities, and behavioral patterns [87]. Such information is obtained either explicitly using online registration forms and questionnaires resulting in static user profiles or implicitly by recording the navigational behavior and/or the preferences of each user resulting in dynamic user profiles [73].

There are different ways to analyze the collected data. Content based filtering methods select content items that have a high degree of similarity to the user's profile [88]. An alternative to content based filtering is the collaborative filtering techniques which allow users to take advantage of other users behavioral activities based on a measure of similarity between them [88; 89]. Rule based filtering allows website administrators/marketers to specify business rules based on user demographics. The rules are used to affect the content introduced to a particular user.

Pattern discovery is the next step of the personalization process. In this step, different data mining techniques, such as clustering, classification, association rule mining, and sequential pattern analysis, are used to discover interesting patterns from web usage data.

Clustering is used to group users with common browsing behavior. The authors in [90] implement a *Profiler* system which captures client's selected links, page order, page viewing time, and cache references. That information is used to cluster users with similar interests. The work in [65] proposes a recommendation engine which considers the association rules between different web pages, and

the derivation of URL clusters based on two types of clustering techniques in conjunction with the active user session. The recommendations are then added to the last requested page as a set of links before the page is sent to the client browser.

Association rules or sequential pattern discovery methods facilitate the identification of related pages or navigation patterns which can be used subsequently to recommend new web pages to the visitors of a website. The work in [64] provides a framework for web personalization based on association rule mining from click-stream data. [91] introduces the *System L-R* recommendation system which constructs user models by classifying the web access and recommends relevant pages to the users based both on the user models and the web content.

The authors in [92] present a web usage mining system *KOINOTITES* which uses web usage mining techniques to identify groups of users who have similar navigation behavior. The produced information can either be used by the administrator in order to improve the structure of the website or it can be fed directly to a personalization model, (e.g., *collaborative filtering*). The work in [93] proposes a web mining strategy for web personalization based on a novel pattern recognition strategy which analysis and classifies users taking into account both user provided data and navigational behavior of the users. It presents the *Referrer Based Page Recommendation, RBPR,* that uses information about a visitor's browsing context (specially, the referrer URL provided by the HTTP) to suggest pages that might be relevant to the visitors underlying information need.

The authors in [94] introduce a different approach of personalization that requires no input or feedback from the user. [88] suggests a set of steps that make the personalization process effective starting from data collection and managements

efforts, to measuring and evaluating the success of personalization.

4.7 Web Content Mining

Web content mining is mining the data that a web page contains. The contents of most of the web pages are texts, graphics, tables, data blocks, and data records. A lot of research has been done to cover different web content mining issues for the purpose of improving the contents of the web pages, improving the way they are introduced to the website user, improving the quality of search results, and extracting interesting web page contents.

The authors in [95] propose the *InfoDiscoverer* system to discover informative contents from a set of web pages of a website according to HTML tag $< table >$ in a web page. The system partitions the web page blocks into either informative or redundant. Informative content blocks are distinguished parts of the page, whereas redundant content blocks are common parts. This approach yields to the increase of the retrieval and extraction precision, and reduces the indexing size and extraction complexity.

A number of methods to help user find various types of unexpected information from his/her competitors' websites are proposed in [96]. The work in [97] presents a framework for mining product reputations on the internet. It automatically collects people's opinions about target products from web pages, and it uses four different types of text mining techniques to obtain the reputation of those products. The research in [98] examines the accuracy of predicting a user's next action based on the analysis of the content of the pages requested recently by the user. Predictions are made using the similarity of a model of the user's interest

to the text in and around the hypertext anchors of recently requested web pages. The authors in [99] propose an algorithm called *MDR* (Mining Data Records in web pages) to mine contiguous and non-contiguous data records. It finds all records formed by table and form related tags, i.e., $< table >, < form >, < td >$ $, < tr >$, etc. Such data records are important because they often present the essential information of their host pages.

4.8 Web Structure Mining

Links pointing to a document indicate the popularity of the document, whereas links coming out of a document indicate the richness or the variety of topics covered in the document. Web structure mining describes the organization of the content of the web where *structure* is defined by *"hyperlinks between pages and HTML formatting commands within a page"* [100].

Understanding the relationship between contents and the structure of the website is useful to keep an overview about websites. The work in [101] describes an approach that allows the comparison of web page contents with the information implicitly defined by the structure of the website. In this way, it can be indicated whether a page fits in the content of its link structure, and identify topics which span over several connected web pages. Thus supporting web designers by comparing their intentions with the actual structure and content of the web page.

Other studies deal with the web page as a collection of blocks or segments. The authors in [102] use an algorithm to partition the web page into blocks, by extracting the page-to-block, block-to-page relationship from link structure and page layout analysis, a semantic graph can be constructed over the WWW such

that each node exactly represents a single semantic topic, this graph can better describe the semantic structure of the web. [100] presents a survey of some of the ways in which structure within a web page can be used to help machines understand pages.

4.9 Discussion

From previous, it is clear that making changes and adaptations to websites with the help of extracted patterns using different data mining techniques is very effective, but doing that in the maintenance phase can be costly and time consuming and suffers from different drawbacks [2]. In commercial companies which are companies that sell different kinds of products on the web, in order to make an effective maintenance to their websites, the companies have to wait some period of time, for example one year, in order to have a representative log file that reflects customers transactions in their website and can give a clear image about their behavior. This amount of time is considered very big especially for the companies in which the time factor plays an important role in their success strategy, and have many competitors who can attract their customers if they have no solid marketing strategies in order to keep their customers as loyal as possible. On the other hand, most businesses gather information about internet customers through online questionnaires. But, many customers choose not to complete these questionnaires because of the amount of time required to complete them as well as a lack of a clear motivation to complete them. Several companies use cookies to follow customers through the WWW, but cookies are sometimes detected and disabled by web browsers and do not provide much insight into customer prefer-

ences. This is because customers are feeling that their profiles are not secure so a number of customers choose to give incorrect information about themselfs.
Furthermore, as discussed previously in section 4.4, in web mining different strategies are implemented to identify sessions such as defining a time threshold that a session should not exceed or assuming that consecutive accesses within the same time period belong to the same session. In some cases, it is difficult to identify users, for example, when two users use the same machine and the same browser with the same IP address and look at the same set of pages [71]. We can conclude from that, that those session and user identification strategies can not give a guarantee that those identified users and sessions represent the actual users and sessions. In contrast to that, in our methodology, we can guarantee that we mine the actual user transactions and profiles that were collected from users personally. For example, in a telecommunication company, when a customer want to sign a mobile telephone contract, he usually fills a form that represents his profile. Any further products or services requested by the user will also be recorded.
In our approach, the problem of building an ill-structured website for some company/business can be solved by applying data mining techniques such as clustering, classification, and association rule mining on the contents of the information system of the company/business. Then, from the extracted patterns, the information needs to be considered in the website building process is gained and invested during the design phase in the process of website design which yields to a better designed retail website. The main advantage of this method is that it reduces maintenance time and budgetary costs for websites if they are built taking into account the extracted interesting patterns from the transactions database of the company/business. Furthermore, in this case, the customers transactions

are more correct and represent the actual customers profiles and behaviors. This approach also permits the sales manager to focus on the core business and gives him a better view about his products and customers which is very helpful in designing retail websites.

Our contribution in improving the structure of websites through the investment of the patterns extracted from different data mining techniques to build well-structured websites during the design phase is the subject of the next chapter.

Chapter 5

Data Mining in the Website Design Phase

As mentioned before, making adaptations and improvements to websites in the maintenance phase can be costly and time consuming. An alternative approach for designing retail websites is improving the structure of the website in the design phase before being published. In our contribution (see [3; 4; 5; 6; 7]), the problem of building an ill-structured website for some company/business can be solved by applying data mining techniques such as clustering, classification, and association rule mining on the contents of the information system of the company/business. Our method is summarized as follows: Suppose that there is a physical shop that sells different kinds of products and introduces some services to its customers who are physically living near the shop. Beside customers profiles, the shop has a database that stores customers transactions in the shop. A customer transaction records what products did the customer buy, what kind of services have been requested by him, and other useful information such as the transaction time. The shop follows specific marketing strategies that meets the requirements of the shop owners and participates in increasing customer satisfaction. The shop wants

to expand its business online in order to reach a bigger sector of customers and overcome time and distance limitations of the physical shop. In our approach, before building the retail website for this shop, we mine customers transactions database using different data mining tasks.

The extracted patterns from mining such databases give valuable information. That information reflects customer behavior, interestingness, product and sales behavior, and other valuable information. Taking into account general requirements of the shop and marketing strategies implemented by the shop owners, the valuable extracted patterns are considered and invested in the process of website design which yields to a better designed retail website. These extracted patterns can be invested as guidelines on how various components of the website should be designed and how they relate to each other.

The main advantage of our contribution is that it reduces maintenance time and budgetary costs for websites if they are built taking into account the extracted interesting patterns from mining the transactions database of the shop/business. It also overcomes the disadvantages of using web mining to improve websites as discussed in Chapter 4. This approach also permits the sales manager to focus on the core business and gives him a better view about his products and customers to improve his marketing strategies which is very helpful in designing retail websites. Our method can be beneficial for many businesses, companies, and commercial institutions such as in the field of retailing, and telecommunication. For example, usually, in the telecommunications business every company has a database that records customers transactions and contains biographic information about its customers such as customer age, sex, marital status, and address. The interesting patterns extracted from mining those information can be effec-

tively invested in the process of designing a website for this company/business and personalize services introduced to customers which will consequently increase the company/business's overall profit.

In the next sections we will see how to use association rule mining, classification, and clustering during the design phase to ensure having a well-structured retail website.

5.1 Association Rule Mining During the Design Phase

Association rule mining is one of the data mining techniques that plays an important role in our approach [8]. An association rule is an expression of the form $X \Rightarrow$Y, where X and Y are sets of items and have no items in common. This rule means that given a database of transactions D where each transaction $T \in$D is a set of items. $X \Rightarrow$Y denotes that whenever a transaction T contains X then there is a probability that it contains Y, too. The rule $X \Rightarrow$Y holds in the transactions set T with confidence c if $c\%$ of transactions in T that contain X also contain Y. The rule has support s in T if $s\%$ of the transactions in T contains both X and Y. Association rule mining is finding all association rules with support and confidence values that are greater than or equal a user-specified $minsup$ and $minconf$ respectively.

In general, the process of extracting interesting association rules consists of two major steps.

The first step is finding all itemsets that satisfy $minsup$ (known as *Frequent-Itemset* generation). The second step, is generating all association rules that

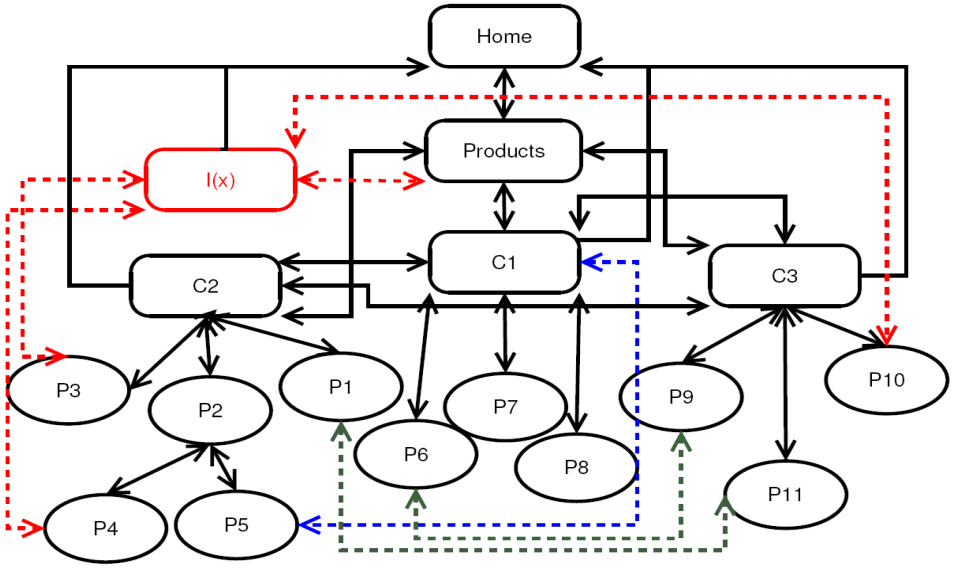


Figure 5.1: Improved Website Design Structure Using Extracted Association Rules

satisfy $minconf$ using itemsets generated in the first step. After generating frequent itemsets, association rules that are greater than or equal to $minconf$ are generated. Those rules are called interesting association rules. Those rules can be invested in many different applications. One of those applications is improving the structure of the company's website that the mined database belongs to. This is done during the website's design phase by creating links between items that seem to be sold together, or highlight those links if they are already exist, and/or create index pages which are pages that have direct links to some products that may be of interest for some group of customers.

Figure 5.1 represent a part of the website's structure of a company that sells different kinds of products. Boxes and circles represent website's pages, and arrows represent links between pages. *C1*, *C2*, and *C3* represent different prod-

uct categories, and *P1*, *P2*, ..., and *P11* represent different products belonging to those categories. The dotted arrows represent links created with the help of the extracted interesting association rules. Note that links between product-to-product, and product-to-category can be created. For example, direct links between both product $P1$ and product $P11$, and product $P6$ and product $P9$ are created depending on some extracted association rules. For example, the link between product $P1$, and product $P11$ is created depending on a rule that says:

```
P1 ==> P11 [support= 4%, confidence= 60%]
```

This rule means that 60% of customers who buy product $P1$, buy also product $P11$ with it and 4% of all customers buy both. Also a link between product $P5$ and category $C1$ is created depending on the rule that says:

```
P5 ==> C1 [support= 5%, confidence= 80%]
```

This rule means that 80% of customers who buy product $P5$ buy also a product belongs to $C1$ category, and 5% of all customers buy both. $I(X)$ is an index page that has direct links to products $P3$, $P4$, and $P10$. That products may be of interest for group X of customers. This index page may be created depending on a set of similar rules such as:

```
age(X, "20...35") ==> P3
age(X, "20...35") ==> P4
age(X, "20...35") ==> P10
```

The previous rules mean that customers who are between 20 and 35 years of old are interested in buying products $P3, P4$ and $P10$ respectively. Consequently, such modifications done to the website's design help customers find their target products in an efficient time, encourage them to buy more from the available

products, and give them the opportunity to have a look at some products that may be of interest for them, which will consequently increase the company's overall profit.

5.1.1 Experimental Work

For the experiments, we used a dataset that represents customer transactions in a grocery store. This dataset consists of 15 attributes. 10 attributes represent the available products: *Readymade, Frozenfood, Alcohol, Freshvegetables, Milk, Backerygoods, Freshmeat, Toiletries, Snacks,* and *Tinnegoods*. The remaining 5 attributes: *Gender, Age, Marital, Children,* and *Working*, represent the gender of the customer, his/her age, his/her marital status, having children or not, and if the customer is a worker or not, respectively. In the mining step, we applied the *Apriori* algorithm implemented within the association rule miner of the *WEKA* tool [20].

Before running any association rule mining algorithm, we designed the prototype of the website that represents the grocery store and the products available in it as shown in Figure 5.2. Of course, this prototype does not represent all products available in the grocery store, it just represents the set of products that are available in the transactions of the tested dataset. A website for the grocery store need to be built in order to give a good view about it, to support and facilitate services introduced to customers, and to encourage customers to buy more from the available products. The website design process starts by identifying the goal of building this website, looking at the transactions database and trying to understand its structure, and checking out what kinds of information are available in the transactions database and whether this information should be presented in

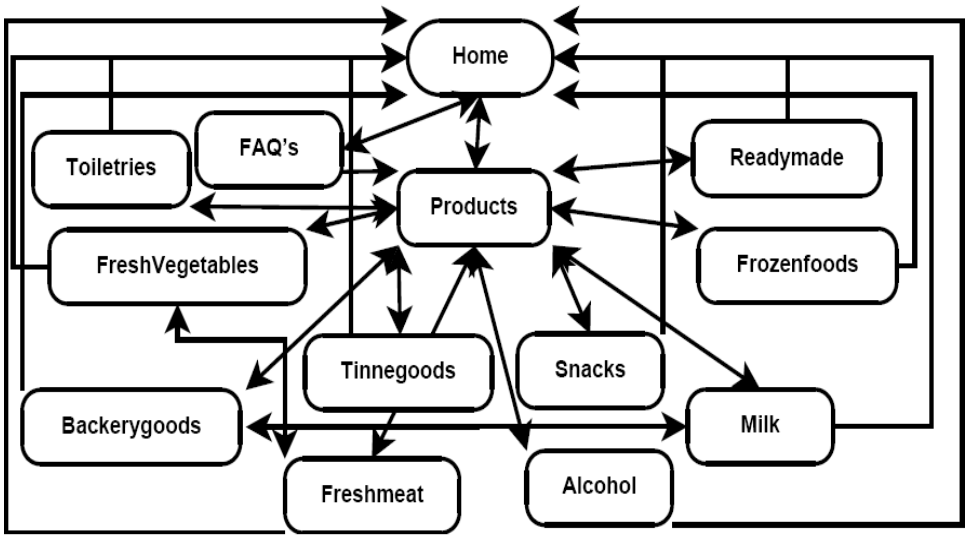


Figure 5.2: Initial Website Prototype of the Grocery Store

the website or not [103]. Beside this information, we considered some standards and recommendations adopted by website designers to have a well-structured website such as every page should be connected directly to the home page, and every parent page should have direct links to its descendants [46]. Products that are expected to be sold together, such as $Freshmeat$ and $Freshvegetables$, are connected directly to each other. As we mentioned before, we used the *Apriori* algorithm discussed in section 2.5.4 to mine for interesting association rules. In the mining process, we used different $minsup$ and $minconf$ values ranging from 8% to 20% for $minsup$, and from 70% to 90% for $minconf$. The best extracted association rules have been studied and analyzed in order to decide how to invest them in the process of designing the website of the grocery store. The website prototype is represented in Figure 5.3. The following rules are an example of the extracted interesting association rules:

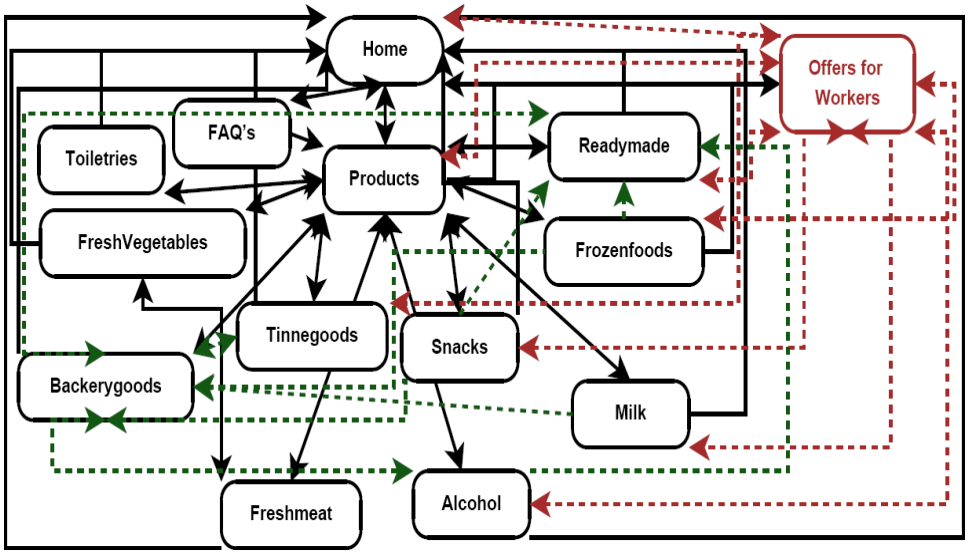


Figure 5.3: Website Prototype With the Help of Extracted AR's

```
1. alcohol=1 milk=1 ==> working=Yes conf:(0.96)
2. alcohol=1 bakerygoods=1 tinnedgoods=1 => readymade=1 conf:(0.81)
```

Both rules are extracted by setting the *minsup* value to be 20% for the first rule and 8% for the second rule. The *minconf* was set to 90% and 80% respectively. The first rule says that 96% of customers who buy *alcohol* and/or *milk* are workers, and 20% of all customers buy *alcohol* and/or *milk* and they are *workers*. From this rule, and other similar extracted interesting association rules, an index page is decided to be created. This index page have direct links to products that may be of interest for the customers who are *workers*. As we see in Figure 5.3, we called this index page *offers for workers*. It has direct links to *Readymade, Frozenfood, Snacks, Alcohol, and Milk*. Those links are represented by the red dashed arrows. The second rule means that 81% of transactions which contain *Alcohol* and/or *bakerygoods* and/or *tinnedgoods* contain also *readymade*, and 8%

of all transactions contain all (i.e *Alcohol*, *bakerygoods*, *tinnedgoods*, *readymade*). From this rule and other similar interesting association rules, direct links from *Alcohol*, *bakerygoods*, and *tinnedgoods* to *readymade* are created. Those links are represented by the green dashed arrows. Some arrows in the prototype are bidirectional, that means that the pair of products connected by a bidirectional arrow are frequently bought together. In other words, customers who buy the first product buy the second product with, and vice versa. For example, customers who buy *backerygoods* buy also *readymade* with, and vice versa. On the other hand, uni-directional arrows employ that customers who buy the first product buy the second product with, but not vice versa. For example, in the prototype of Figure 5.2, a bidirectional arrow between *bakerygoods* and *milk* was created because we believed that those two products are strongly related to each other, so they needed to be connected. But from the extracted rules, we found that a percentage of customers who buy *milk* buy also *bakerygoods*. In contrast, we found no rule which indicates that the customers who buy *bakerygoods* buy also *milk* with. From that, we modified the arrow between those two products to be one directional arrow from *milk* to *bakerygoods*.

As a result, such interesting association rules can be used to design a well-structured website, plan marketing and advertising strategies which will consequently increase the grocery store overall profit. Furthermore, the main advantage of our method is that it reduces maintenance time and budgetary costs for websites if they are built taking into account the associations between different products, and customer buying habits that can be found in the transactions database in almost every shop or grocery store. It also permits the sales manager to focus on the core business and gives him a better view about his products

and customers which is very helpful in designing retail websites. This method also participates in improving customer satisfaction and encourages him to be a frequent buyer.

5.1.2 Method Evaluation

In order to evaluate our method we implemented a simulation tool in *Java*. This tool simulates the behavior of customers in both website prototypes which are the *standard* website prototype which is built without taking into account extracted interesting association rules and the *improved* website prototype which is built taking into account extracted interesting association rules. We divided our dataset into two parts. The first part was used to extract association rules that have been later used in improving the elementary website prototype as discussed in section 5.1.1. The second part was used in the testing process. In every transaction in the dataset, the customer has a list of products he wants to buy. Those products are considered as to be the set of target products. Every product is represented by a page in the website prototype. In the elementary prototype in Figure 5.2, the customer starts at the home page. Then, he starts searching for his first target product. The first target product is chosen randomly from the set of target products presented in every transaction. So, he has to go to *products* page and from there he searches for his first target product. After that, if there is a direct link from that target product page to one of the next target product pages, the new target product page is visited, otherwise he has to backtrack to *products* in order to search for the next target products. This process is repeated until all pages of target products are found.

In the prototype in Figure 5.3, the customer starts at the home page. Then,

before starting to search for his target products, it is checked if the customer is a *Worker* or not. If he is a *Worker*, then he goes directly to *offers for workers* page. From there, he will start searching for his target products. If there exist a direct link to a product page of a certain product in the set of target products, then it would be followed. If there is a direct link from that target product page to one of the next target product pages then the new target product page is visited, otherwise the customer have to backtrack to *offers for workers* in order to search from there for the next target products. If there are no products of the set of target products presented in the *offers for workers*, then the customer backtracks to *products* in order to search for the rest of his target products until all pages of target products are found.

In every step in the process of searching for target products, all existing links Li from any visited page $i = 1, 2, ..., m$ to any other pages are calculated. We defined the cost C_T of every prototype to finish visiting product pages in some transaction T to be the ratio between the sum of all existing links that may be visited in every transaction $\sum_{i=1}^{m} Li$, and the number of target pages (i.e. products) n, where $n \neq 0$:

$$C_T = (\sum_{i=1}^{m} Li)/n$$

The lower the value of C_T is, the higher the efficiency of the prototype is to find target products in transaction T. In other words, the customer will need less time and effort in order to reach his target products as the value of C_T becomes lower. We ran the simulation tool at both prototypes simultaneously. We assumed that we have a total of 100 products in the grocery store. Figure 5.4 shows the average costs of both prototypes to finish 500 transactions. The columns represent the average costs with respect to different number of products in both

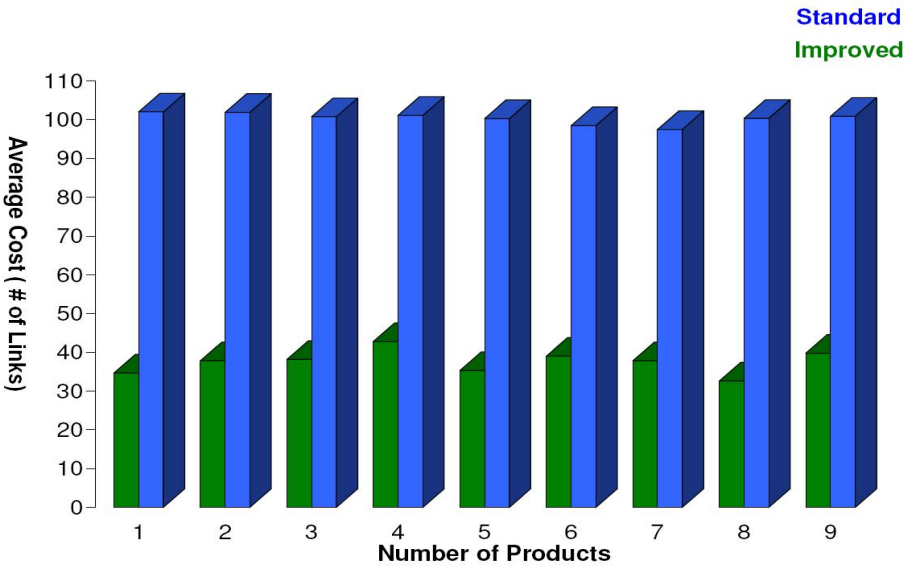


Figure 5.4: The Average Costs of Both Prototypes

prototypes. The blue columns (titled *Standard*) represent the average costs of the elementary prototype in Figure 5.2. The green columns (titled *Improved*) represent the average costs of the *improved* prototype in Figure 5.3. In the *improved* prototype, the average cost is reduced to 62% in comparison to the average cost of the *standard* prototype. Our method reduced the cost in the *improved* prototype up to 90% for some transactions in comparison with the costs of the *standard* prototype.

5.2 Classification and Clustering During the Design Phase

Clustering and classification are two data mining techniques that play an important role in our methodology [4; 6; 7]. Clustering is the process of partitioning a given population of events or items into sets of similar elements, so that items

within a cluster have high similarity in comparison to one another, but are very dissimilar to items in other clusters [12]. In web usage mining there are two main interesting clusters to be discovered: usage clusters and pages clusters [52]. The authors in [56] present an approach to cluster web pages to obtain high quality clusters of web pages and use those clusters to produce index pages, where index pages are web pages that have direct links to pages that may be of interest of some group of website navigators. In [57] clustering techniques are applied to web log file to discover those subsets of web pages that need to be connected and to improve the already connected pages. The authors in [33] use a clustering algorithm which identifies groups of similar sessions, allowing the analysis of visitor behavior.

In contrast to clustering, classification is dividing an existing set of events or transactions into other predefined sets or classes based on some characteristics. In web usage mining, classification is used to group users into predefined groups with respect to their navigation patterns in order to develop profiles of users belonging to a particular class or category [52]. The work in [60] proposes an approach to reorganize a website based on user access patterns and the classification of web pages into two categories: index pages and content pages. [91] introduces the *System L-R* recommendation system which constructs user models by classifying the web access and recommends relevant pages to the users based both on the user models and the web content. In [93], the authors propose a web mining strategy for web personalization based on a novel pattern recognition strategy which analysis and classifies users taking into account both user provided data and navigational behavior of the users. They present the *Referrer Based Page Recommendation, RBPR,* that uses information about a visitor's browsing con-

text (specially, the referrer URL provided by the HTTP) to suggest pages that might be relevant to the visitors underlying information needs.

In this section, we will see how we apply clustering and classification data mining tasks to support retail website design during the design phase as discussed in the previous section.

5.2.1 Experimental work

For the experiments, we used a dataset that represents customer transactions in an electronics store. This dataset contains information about customers profiles, transactions, purchases, and store branches. A website for this company is decided to be built to give a good view about its work, to support and facilitate services introduced to customers, and to encourage customers to buy more from its products. The website design process starts by identifying the store's goal from building this website. The website designer looks at the dataset and try to understand its structure, check out what kind of information is available in the dataset and whether those information should be presented in the website or not [103]. Beside that information the website designer considers some standards and recommendations adopted by website designers to have well-structured websites such as every page should be connected directly to the home page and every parent page should have direct links to its descendants [45; 49]. Figure 5.5 shows a screen shot of the resultant website design.

The website is well-designed with respect to website designers standards. Every page has a consistent navigation link to move around in the website. When a user logs in, a number of random products are offered to him. When he visits some product page, a list of related products are introduced to him. Those

Figure 5.5: Standard Website Design

products are related to the current product with respect to product type and manufacturer.

For example, in Figure 5.5, the customer choosed to buy *Canon Digital Camera*. From that, a list of other cameras and related products from the same manufacturer and other manufacturers are presented to the customer in order to be able to compare the same product from different manufacturers with respect to different attributes. To improve the design of this retail website, we used different classification and clustering algorithms implemented in the data mining tool WEKA [20] to mine the transactions dataset of the electronics store. For classification we used the *ADtree* and the *J48* algorithms. The *ADtree* is an implementation

of Freund and Mason [104] Alternating Decision Tree Algorithm. In addition to classification, the *ADtree* gives a measure of confidence called the classification *margin* [105]. *J48* algorithm is an implementation of the *C4.5* decision tree learner [25]. The algorithm uses the greedy technique to induce decision trees for classification. A decision-tree model is built by analyzing training data and the model is used to classify unseen data. For clustering we used the *SimpleKmeans* algorithm. The *SimpleKmeans* algorithm clusters data using the *K-Means* algorithm. The *K-Means* algorithm takes the input parameter k, and partitions a set of n objects into k clusters so that the resulting similarity within the cluster is high but the similarity with other clusters is low. Cluster similarity is measured in regard to the mean value of the objects in the cluster [12]. In the experiments we used different numbers of clusters to be generated ranging from 2 to 10 clusters. After tuning the number of generated clusters, we found that 5 clusters is the most suitable number of clusters as it can give very descriptive results. After running the above mentioned algorithms, a lot of interesting patterns have been extracted. In all of the above algorithms, the extracted patterns were saved for visualization in order to ease the process of data analysis.

We summarized those interesting extracted patterns in Figure 5.6. As we see in the flowchart in Figure 5.6, we divided the customers with respect to their biographic information: sex, age, and address into five classes. The first class is the *Male* customers living in *Hamburg* (Cluster0). The second class is the *Male* customers living in *Munich* (Cluster1). The third class is the *Female* customers who are living in *Hamburg* and $\geq$ 36 years old (Cluster2). The fourth class is is the *Female* customers who are living in *Hamburg* but less than 36 years old (Cluster3). The last class is the *Female* customers living in *Munich* (Cluster4).

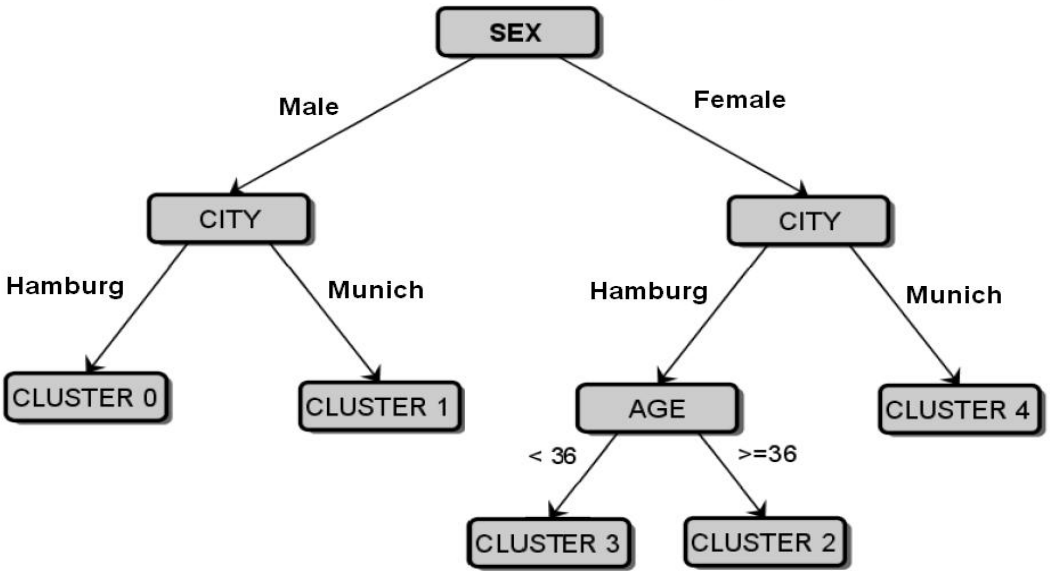


Figure 5.6: A Summary of the Interesting Extracted Patterns

Each class has a list of products that may be of interest for its members. The products are ordered in the list with respect to buying frequency. The most frequently bought product is the first one in the list. On the other hand, the least frequently bought product is the last one in the list. In order to improve the standard website design in Figure 5.5, we took into account the previously extracted patterns in the design process. Figure 5.7 shows a screen shot of the improved website design. After the customer logs in, depending on his biographic information, he is assigned to one of the clusters mentioned above (see Figure 5.6).

From that, a list of products that may be of interest for him are introduced to him. Through this process, we encourage the customer to buy more from the available products, give him the opportunity to have a look at some products that may be of interest for him. Furthermore, we ease the process of product search so that the customer can reach his target product in an efficient time and the

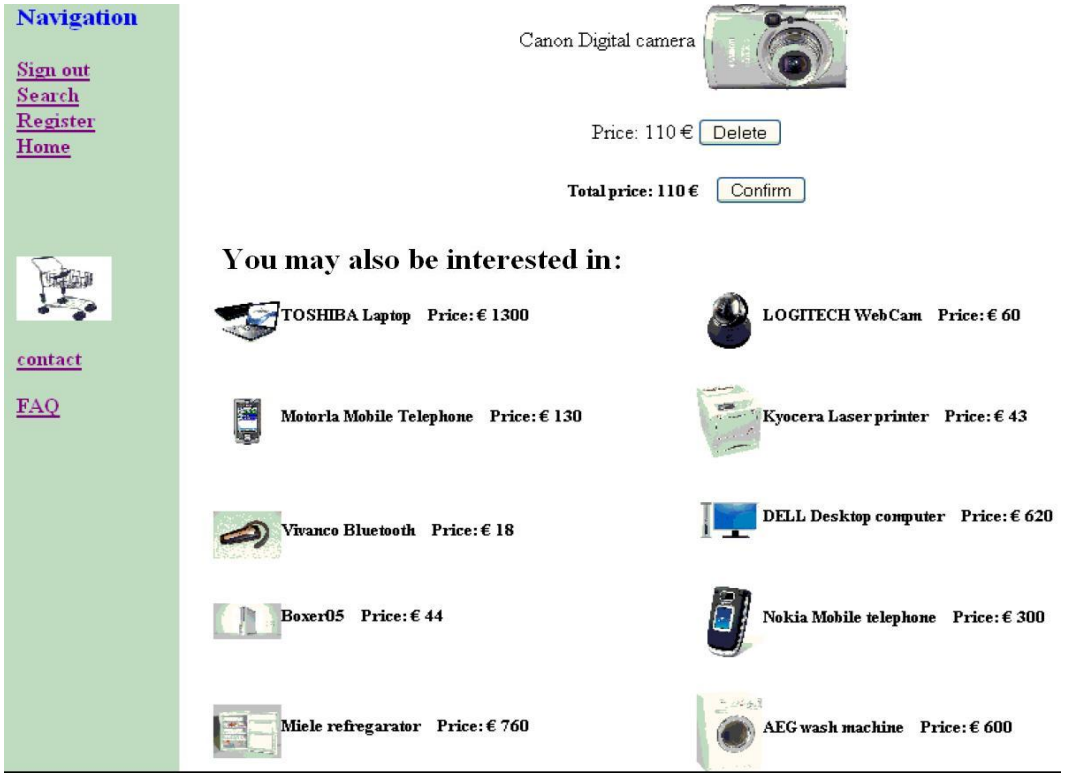


Figure 5.7: Improved Website Design

navigation process will be optimized. This factor is very important in winning customers loyalty and changing them to be frequent buyers. This is because, in most cases, if the customer did not find his target product in a specific time, he will give up and starts to search for another shop or provider in which he may find his target product(s) in.

As a result, such interesting extracted patterns from clustering and classification can be used to design a well-structured retail website, plan marketing and advertising strategies which will consequently increase the electronics store overall profit.

5.2.2 Method Evaluation

In order to evaluate our method we implemented a simulation tool in *PHP*. The tool simulates the behavior of customers in both websites. We call the first website the standard website, which is built without taking into account patterns extracted from the electronics store transactions dataset using clustering and classification. On the other hand, the improved website is the one built taking into account interesting extracted patterns from the electronics store transactions dataset using clustering and classification techniques. In every transaction in the dataset, the customer has a list of products he wants to buy. Those products are considered as to be the set of all target products. Every product is represented by a page in both websites. We defined the session time $Time_c$ for some customer C to be the amount of time needed to buy all target products as follows:

$$Time_c = \sum_{i=1}^{m} \underbrace{(x_1 \cdot t_1)}_{\mathbf{X}} + \sum_{i=1}^{n} \underbrace{((x_2 \cdot t_1) + t_2)}_{\mathbf{Y}}, \text{ where}$$

X: The total time needed to reach a product that belongs to the offered list of products

Y: The total time needed to reach a product that does not belong to the offered list of products, in addition to the time needed to search for this product

m: Number of products that belong to the offered list of products

n: Number of products that do not belong to the offered list of products

x_1: Number of web pages needed to reach the web page of a product

belonging to the offered list of products

x_2: Number of web pages needed to reach the web page of a product not belonging to the offered list of products

t_1: Time needed to load a web page

t_2: Time needed to fill the search form

For example, if some customer bought three products, two product belong to the offered list of products, and one product is outside the list of offered products. The overall time will be calculated as follows. Two seconds to load a web page (t_1) and 7 seconds to fill the search form (t_2). We set x_1 to be 1 (i.e. the customer need one web page to reach a product belonging to the offered list of products) and x_2 to be 3 (i.e. the customer needs three web pages to reach a product not belonging to the offered list of products). Then the overall time will be calculated as follows:

$$Time_c = \sum_{i=1}^{2}(2 \cdot 2) + \sum_{i=1}^{1}(3 \cdot 2) + 7) = 21 \ Seconds$$

Measuring the session time starts from customer *log-in* time until buying the last product by clicking on *Confirm* button. Figure 5.8 shows the average session time needed to finish all customer transactions with respect to customers clusters. The figure shows a clear difference between the session average times in both standard and improved websites. The sessions average time of the improved website is reduced to 51% of the sessions average time of the standard website. Table 5.1 is a comparison between the efficiency of both websites with respect to transaction times where:

```
A: Number of transactions where Time(standard) > Time(Improved)
B: Number of transactions where Time(standard) = Time(Improved)
C: Number of transactions where Time(standard) < Time(Improved)
```

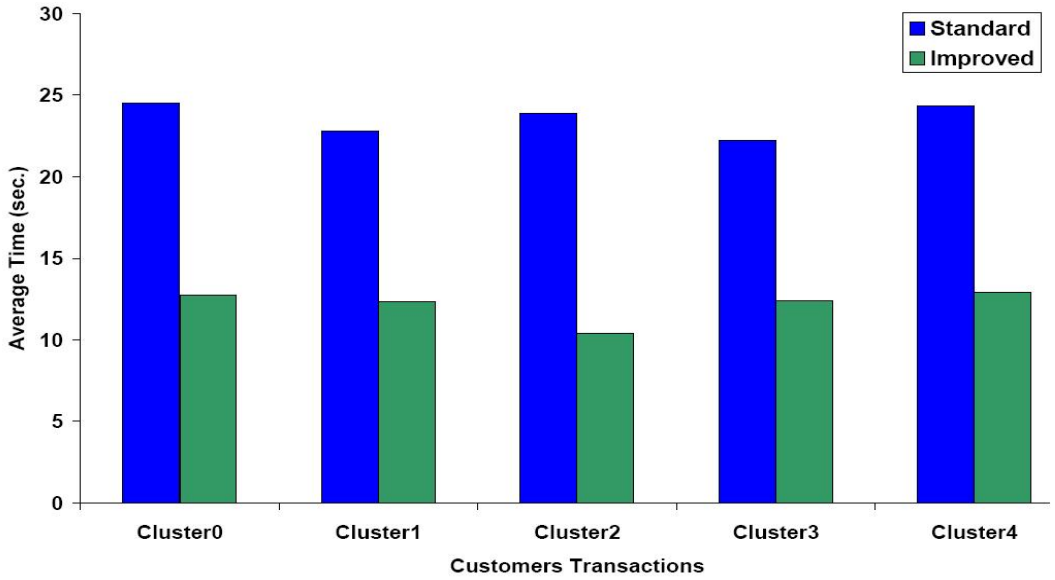


Figure 5.8: Average Session Times Needed to Simulate Customers Transactions in Both Websites

Beside evaluating the efficiency of both websites by simulating customer behavior in them, we made a manual evaluation by allowing a number of persons to navigate in both websites. A random of 50 transactions have been taken from the transactions dataset. Every person has been given a list of target products he need to buy. In this case, the target products are the products that belong to a specific transaction of the chosen 50 transactions. Then, we allowed the users

Table 5.1: A Comparison of Session Times Within Different Clusters

Cluster	*A*	*B*	*C*
0	385	149	13
1	359	37	1
2	137	9	0
3	301	37	1
4	385	52	3

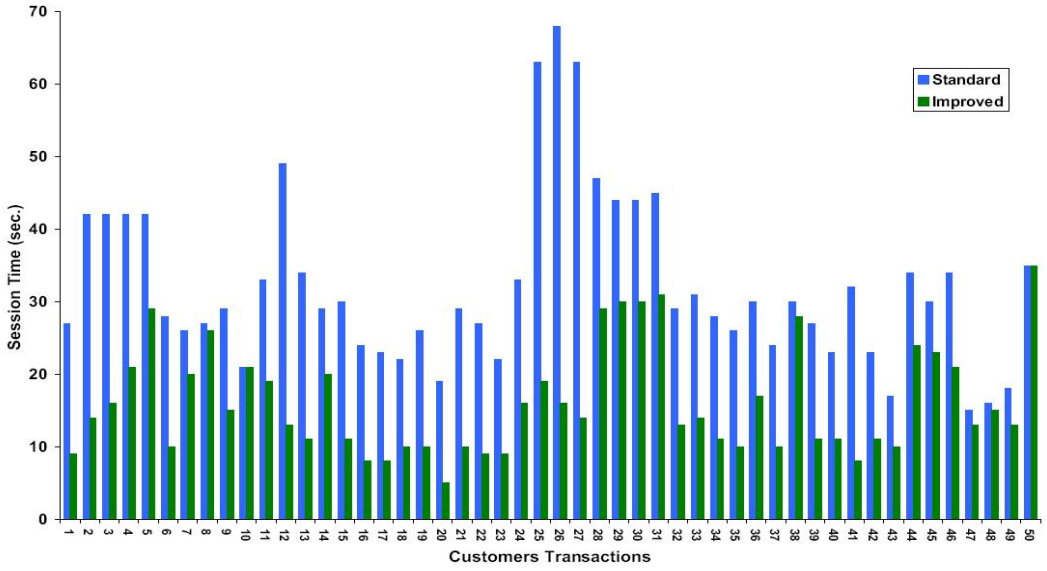


Figure 5.9: Time Needed to Finish 50 Transactions Manually

to use both websites (standard and improved) in order to search for their target products. After that, the sessions times were measured as previously mentioned for both websites. Figure 5.9 shows the resulted navigation time in both websites to complete the sessions of 50 transactions.

5.3 Datasets Availability

One of the main problems we faced from the beginning of this dissertation work, is finding a suitable dataset that represents customers transactions in some company, store, or business to run our experiments. Even though there are a lot of internet resources that provide repositories of sample datasets such as the *UCI Knowledge Discovery in Databases Archive*[1], those datasets are mostly synthetic datasets, do not provide enough information that meets our experimental needs,

[1] `http://kdd.ics.uci.edu/`

and as far as we know there is no dataset presented in such repositories represents a real dataset of customers transactions in a physical store or business. This is because most businesses do not want to make their data public. On one hand, for the purpose of preserving customers privacy. On the other hand because they do not want their competitors see their data and know how their business is running. One of the available solutions for such problem is the idea of synthetic dataset generators which will be discussed in detail in Chapter 7.

Chapter 6

Temporal Frequent Itemset Mining for Enhanced Marketing

Temporal data mining is the process of mining databases that have time granularities [106]. The goal is to find patterns which repeat over time and their periods, or finding the repeating patterns of a time-related database as well as the interval which corresponds to the pattern period [107]. In section 6.2, beside the usage of interesting association rules, we will see how we use the association rules that do not satisfy minimum requirements (i.e. have support and confidence values less than the user specified $minsup$ and $minconf$ respectively) in the decision making process [8]. Then, in section 6.3, we present a new method to mine for interesting frequent itemsets. Our method based on the idea that interseting frequent itemsets are mainly covered by many recent transactions [9].

6.1 Related Work

Association rules mining techniques are used in web usage mining to find pages that are often viewed together, or to show which pages tend to be visited within

the same user session [61]. The authors in [63] propose an approach for predicting web log accesses based on association rule mining. In web usage mining, sequential pattern mining could be used to predict user's future visit behaviors. [68] suggests using adaptive websites to attract customers using sequential patterns to display special offers dynamically to them. The work presented in [108] studies the problem of association rules that exist in certain time intervals and thus display regular cyclic variations over time. In [109] the authors present an algorithm that utilizes the transaction time interval of individual customers and that of all customers to find out when and who will buy products, and what items of products they will buy.

The work in [110] presents a visualization technique to allow the user to visually analyze the rules and their changing behaviors over a number of time periods. This enables the user to find interesting rules and understand their behavior easily. This approach differs from our approach in that it does not take into account in the decision making process rules that are semi-interesting, that either could be a part of the set of interesting association rules in the next time period, or they were a part of that set in the last period of time. As far as we know, there is no previous work that takes semi-interesting association rules into account in the process of analyzing association rules.

6.2 Periodical Association Rule Mining

Assuming that we have a physical store which has a dataset of customers transactions and that the store sells also its products online through its own website, and we have a log file that records users sessions and navigations in the online

store, our method is summarized as follows.

1. Periodical mining for association rules. The period length is user specified. It depends on different factors such as the type of products available in the store, the expected time of the change of users behavior, how often we add new products to the products set, or remove some products from that set. For example, a store that has frequent addition/removal of products should have a shorter time period than that of stable products.

2. From the first step we get three types of association rules:

   - **Interesting Association Rules** which are rules that have support and confidence values greater than or equal $minsup$ and $minconf$ respectively.
   - **Semi-interesting Association Rules** which have support and confidence values less than the user-specified $minsup$ and $minconf$ respectively. But their values are close to $minsup$ and $minconf$ values. For example, if the $minsup$ is 10% and we have an association rule that has a 7% support value, then this rule is considered a semi-interesting association rule.
   - **Non-interesting Association Rules** which are rules that have support and confidence values much lower than the user-specified $minsup$ and $minconf$ values respectively. For example, if the $minsup$ is 10% and we have an association rule that has a 2% support value, then this rule is considered a non-interesting association rule. Those rules represent products that have weak sales.

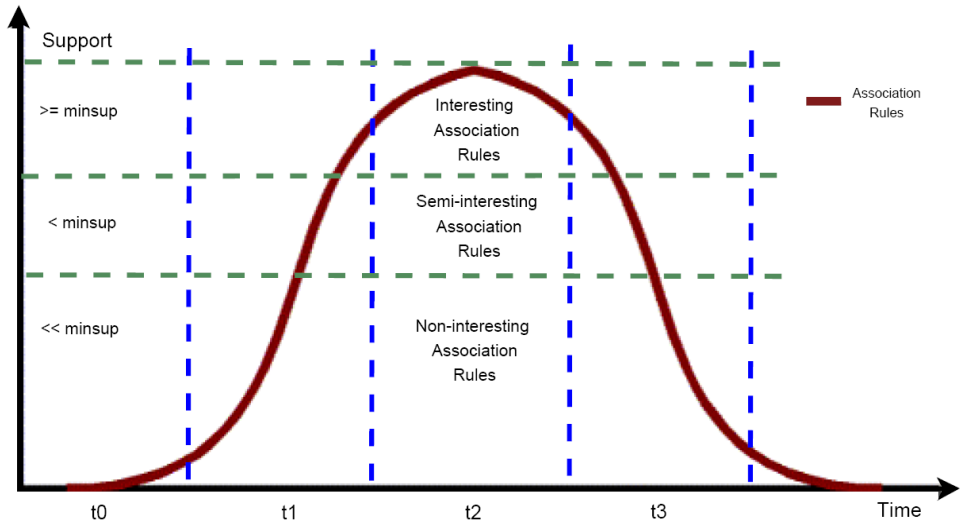


Figure 6.1: The Interestingness of Association Rules of a Set of Products With Respect to Different Time Periods

3. Use these three types of association rules to make decisions. Interesting association rules give an idea about the relationship between different products. For example, how often are two products sold together. From that we can, for example, make offers/discounts in order to encourage customers to buy from those products. Figure 6.1 represents the behavior of association rules of a set of products through different time periods. X-axis represents the time in which association rules through that set of products, during *t0, ..., t3* time periods, are extracted. Y-axis represents the interestingness of association rules represented by different support values. The first sector represent non-interesting association rules represented by $<< minsup$. The second sector represents semi-interesting association rules represented by $< minsup$. The upper sector represents interesting association rules and represented by $>= minsup$.

t0 is the first time period in which we start to sell that products. We see no interesting association rules in that period of time. Then, in *t1* time period, we get three types of association rules: non-interesting, semi-interesting, and interesting association rules. Non-interesting association rules represent the products that still have weak sales. Usually, the sales of some products are weak because either they are recently added to the products set, or the customers are not familiar with such products. In both cases, more promotions need to be made in order to increase the sales of those products taking into account the marketing strategies followed by the store. If the products have interesting association rules between them, then they are considered well-sold products.

Semi-interesting association rules represent the products that start to have good sales but they are not good enough to be considered well-sold products. This gives also an indication that those products may be a part of the well-sold products set in the next time period. In order to make them well-sold products, we can improve, adjust, or implement new marketing strategies, and make promotions or offers for those products to encourage customers to buy more from them which may bring that products to the set of well-sold products. The curve in the third time period *t2* represents intersting association rules. The goal is to keep the products in the set of well-sold products as long as possible.

In the fourth time period *t3*, the products start to have decreased sales because of some reasons, for example the availability of competitive products in other companies. To solve this problem, we can improve the sales of those products by making good offers for those products in a way that attracts

the customers to our products and keep them away from other competitive products which will bring our products back to the well-sold products set and keeps them in that set a longer time.

In that way we give the sales manager a better view about his products and their behavior, and help him to make right decisions and better marketing strategies. We can also predict and control the next best sales which will consequently increase the store's overall profit.

Our method can be applied to both the transactions database of the store and to the log file of the website of the store in the case that the store has its own website and sells its products online. In the case of mining the log file of the store's website, we can use the extracted association rules to improve/maintain the structure of the website, improve the way that some products are presented to the customer, invest the store website to make offers and promotions, and improve the marketing strategies to meet the basic requirements of the store.

6.3 Temporal Frequent Itemset Mining

A fundamental problem for mining association rules is mining frequent itemsets. In a market basket transaction dataset, frequent itemset mining is the process of searching for itemsets that a set of customers likely to purchase in a given visit to the store. In our approach [7; 9], we study the behavior of frequent itemsets with respect to time. This is done through mining for frequent itemsets in different time periods. Top level goal of our temporal analysis is to filter for interesting frequent itemsets. We argue that interesting frequent itemsets are mainly covered

by many recent transactions.

Supposing that we have a dataset that represents the log of some website. The web log consists of a set of records R that represent user requests. Every request consists of a client IP address, the requested URL, and the time stamp t, when the URL is requested. Most websites implement user tracking in some way, mostly using cookies. This makes it possible to map the client IP address to some user ID, which is unique for a user. This avoids difficulties with users, whose IP address changes over time. A session is created by taking consecutive requests of a particular user with small waiting time t_{wait} between the consecutive requests and a maximum session time length $t_{\max}$. We consider the time stamp of the first request of a session to be the time stamp of the session. As a result, we get the session which is a sequence of visited URL's with a time stamp. The set of all sessions S may include multiple sessions of the same users. The set of unique URL's of all sessions in S forms the set of items I used for frequent itemset mining in the next step. The sessions in S are transformed to transactions by removing the duplicate URL's which may appear several times during a single session. So, we have a one to one mapping from the set of sessions S to the set of transactions T.

The set FI include all frequent itemsets $f \subset I$, which are covered by more than $minsup$ transactions. The frequent itemsets can be found from T by the *Apriori* algorithm. Those frequent itemsets represent URL's that are frequently visited together by the users.

In order to include the temporal information into our analysis, for each fre-

quent itemset $f \in FI$, the set of covering transactions $cover(f)$ are found:

$$f \in FI, \;\; cover(f) = \{t: t \in T, f \subset t\} \tag{6.1}$$

Since a transaction $t \in T$ corresponds to exactly one session in $s \in S$, we can trace back the time stamp of s once we know t. So, we construct a time series for each frequent set $f \in FI$ as follows. First, all covering transactions $cover(f)$ are determined for f. Second, the set of time stamps of the session corresponding to all $t \in cover(f)$ is found. Then, the found set of time stamps is ordered and forms the time series s_f corresponding to the frequent itemset f. Note that the length of s_f equals the support of f:

$$f \in FI, \;\; length(s_f) = support(f) \geq minsup \tag{6.2}$$

The set of time series for all frequent itemsets is denoted by TS.

Definition: A frequent itemset A is interesting if $t_c - median(s_A) \leq \delta$ where t_c is the current time, s_A is the ordered time series of the frequent itemset A, and δ is a positive number that represents a time threshold.

That means finding all frequent itemsets from the transactions within time interval $[t_c - \delta, t_c]$ with respect to minimum support $minsup/2$ gives a super set of all interesting frequent itemsets. Figure 6.2 shows an example of interesting and non-interesting frequent itemsets. The frequent itemsets $\{b, d\}$ and $\{a, b, c\}$ are not interesting because the *median* is outside the time interval $[t_c - \delta, t_c]$. It is also clear that the frequent itemset $\{a, b, c\}$ is too old to be interesting. On the other hand, $\{a, e, f\}$ is interesting because the *median* of its time series is within the determined interval. A neccessary condition for interesting frequent itemset A is:

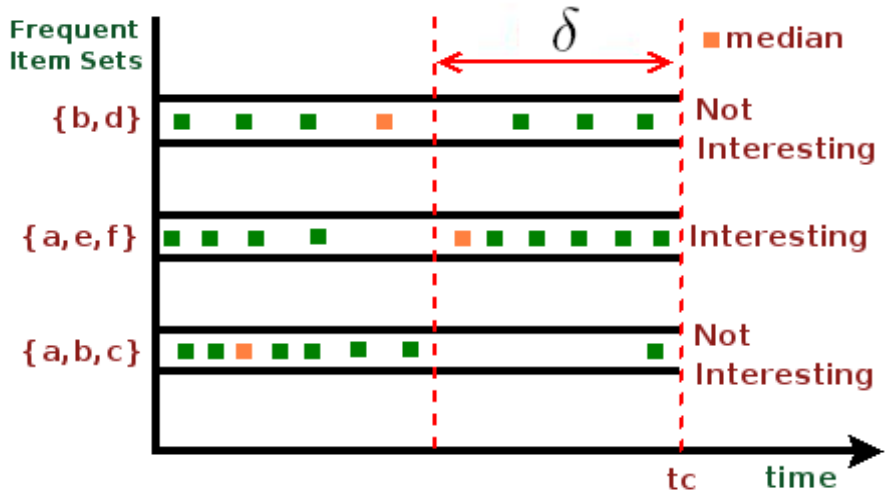


Figure 6.2: Interesting and Non-interesting Frequent Itemsets

$t_c - Q_p(A) \leq \delta$ with $p = 1 - \frac{minsup}{2 \times sup(A)}$ where $Q_p(A)$ is the Quantile p of the time series of set A.

The $quantile_p$ of the ordered set A of values is a number x such that a proportion p of the set are less than or equal to x. For example, the $quantile_{0.25}$ (called also the *quartile*) of an ordered set of values is a value x such that 25% of the values of the set are below that value, and 75% of the values are above that value. The $quantile_{0.5}$ (same as the *median*) is the central value of the set, such that half the values are less than or equal to it and half are greater than or equal to it.

The usage of the above condition means finding all frequent itemsets from the transactions within the time interval $[t_c - \delta, t_c]$ with respect to minimum support $minsup' = minsup/2$, gives a super set of all frequent itemsets. This is because when the frequent itemset A is interesting, then $t_c - median(s_A) \leq \delta$, and because the *median* is the central values of the frequent itemset, then we need at least half of the interval $[t_c - \delta, t_c]$. If the itemset is interesting, then it should definitly be

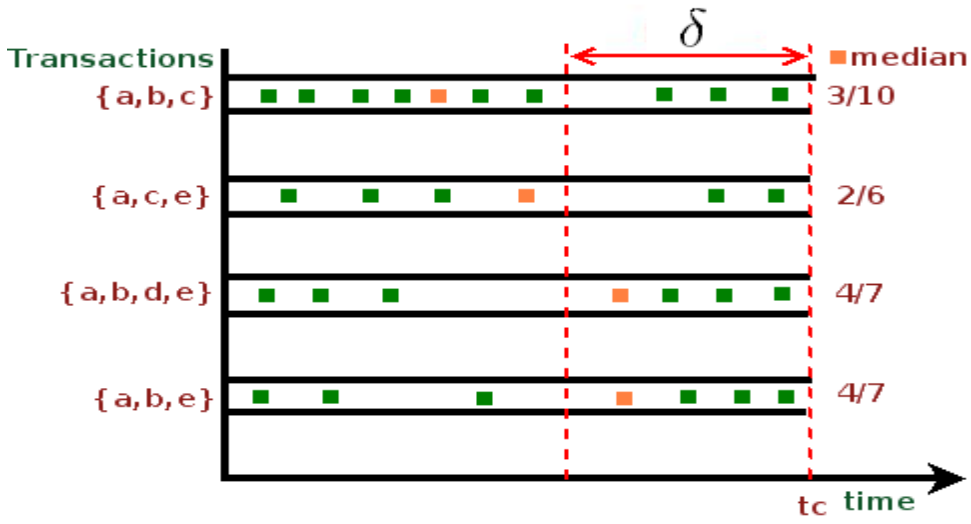


Figure 6.3: An Example of Using the Necessary Condition

frequent. Therefore, $minsup/2$ supporting transactions should be in the interval $[t_c - \delta, t_c]$. This condition can be used either as a preprocessing step to search for frequent itemsets within the determined interval, or as an extension to the *Apriori* algorithm to prune non-interesting frequent itemsets. Figure 6.3 is an example of applying this condition. In this example, we have the $minsup = 10$. When using this condition, the search for frequent itemsets will be within the interval $[t_c - \delta, t_c]$ with $minsup' = 10/2 = 5$. From that, we get a super set of all interesting frequent itemsets. The 3-candidate frequent itemset with $minsup' = 5$ is {a,b,e}. On the other hand, the 3-candidate frequent itemsets with $minsup$=10 are {a,b,c} and {a,b,e}. Candidates for interesting frequent itemsets are (interesting frequent itemsets are underlined):

{a}, {b}, {c}, {e}

{a,b}, {a,c}, {a,e}, {b,e}

{a,b,e}

On the other hand, all frequent itemsets are :

{a}, {b}, {c}, {e}

{a,b}, {a,c}, {a,e}, {b,c}, {b,e}

{a,b,c}, {a,b,e}

Using this method, we reduced the cost of searching for frequent itemsets. This method can be used to improve the search strategy implemented by the *Apriori* algorithm. A time series s_f of a frequent itemset f is an ordered sequence of time stamps of the covering transactions. This set will be helpful in finding out what kind of changes are occurring in which time periods. This can give an indication of the behavior of users with respect to time. We can find why some pages are frequently visited and why others not. Through that we can get a better view about our pages, and also about the users visiting our website. We can predict the future behavior of users depending on the periodical behavior of the users that we have already extracted.

6.4 Experimental Work

For the experiments we used a dataset that contains the preprocessed and filtered sessionized data for the main DePaul CTI Web server [1]. The data is based on a random sample of users visiting this site for a 2 week period during April of 2002. Each session begins with a line of the form:

```
SESSION #n (USER_ID = k)
```

where n in the session number, and k is the user id. There may be multiple consecutive sessions corresponding to the same user id (repeat users). Each sessions

[1] http://www.cs.depaul.edu

ends with a "dashed line". Within a given session, each line corresponds to one pageview access. Each line in a session is a tab delimited sequence of 3 fields: time stamp, pageview accessed, and the referrer. The time stamp represents the number of seconds relative to January 1, 2002.
In order to illustrate what we made in our experimental work, let us take a sample of three sessions that could be found in the web log file.

```
SESSION #1  (USER_ID = 11)
9374553 /news/default.asp  /news/
9374590 /people/search.asp?sort=pt  /news/default.asp
9374610 /people/facultyinfo.asp?  /people/search.asp?sort=pt
9374685 /news/default.asp  /people/facultyinfo.asp
9374720 /courses/  /news/default.asp
------------------------------------------------------------
SESSION #2  (USER_ID = 22)
9185108 /admissions/  /programs/
9185138 /news/default.asp  /news/default.asp
------------------------------------------------------------
SESSION #3  (USER_ID = 33)
9226945 /people/search.asp?sort=pt  /people/
9226975 /people/facultyinfo.asp?  /people/search.asp?sort=pt
9227072 /advising/  /courses/
9227098 /people/search.asp?sort=pt  /news/default.asp
```

After preprocessing which includes the removal of redundant *URL*'s within every session, considering the time stamp of the first request in every session as the time stamp of the session, and the removal of the referrer we get:

```
SESSION #1 (USER_ID = 11)
9374553   /people/search.asp?sort=pt
          /people/facultyinfo.asp
          /news/default.asp
```

```
          /courses/
-----------------------------------
SESSION #2 (USER_ID = 22)
9185108   /admissions/
          /news/default.asp
-----------------------------------
SESSION #3 (USER_ID = 33)
9226945    /people/search.asp?sort=pt
           /people/facultyinfo.asp
           /advising/
           /news/default.asp
```

Then we build the set of items which consists of the set of unique *URL*'s of all sessions:

```
Set of items I = {/people/search.asp?sort=pt, /people/facultyinfo.asp,
   /news/default.asp, /courses/, /admissions/,  /advising/}
```

We then used the *Apriori* algorithm to mine for frequent itemsets. We set the minimum support to be 20% which is equivalent to minimum support count equal to 2.

```
Candidate 1-itemset = {
  /news/default.asp (support count =3),
/people/search.asp?sort=pt( support count =2),
/people/facultyinfo.asp(support count =2)}

Candidate 2-Itemset = {
{/news/default.asp, /people/search.asp?sort=pt}(support count = 2),
{/news/default.asp, /people/facultyinfo.asp}(support count =2),
{/people/search.asp?sort=pt, /people/facultyinfo.asp}
(support count= 2) }

Candidate 3-Itemset = {
```

```
/news/default.asp, /people/search.asp?sort=pt,
/people/facultyinfo.asp(support count= 2) }
```

Candidate 3-Itemset represents the frequent itemset f. The set of covering transactions for the frequent itemset f are in session #1 and session #3 which have the time stamps 9374553 and 9226945 respectively. The time stamps are then ordered to get the time series corresponding to the frequent itemset f:

s_f={ 9226945, 9374553}

The experimental results we have got were not representative enough to reflect the applicability of our approach. Such bad results were expected because the dataset we used was not the one we need. On one hand, because the time interval of the transactions is too small (2 weeks) which is not enough to test our ideas. On the other hand, the transactions represent user sessions in a university website, and usually the transactions recorded in the web log file of university websites are not more than course registration, search for an assignment, etc. The dataset we need to run an effective experiment should have a big time interval for example a dataset that represents customers transactions in a retail website within two years. As far as we know such dataset does not exist especially when we talk about time stamped datasets. This lack of such datasets amplifies the need to develop time stamped transactional datasets generators which is the subject of the next chapter.

6.5 Application Fields

This method can be applied in different fields. One application field is in search engine log files for example to find out the most frequently searched keywords in

the last time period. Another application field is in web usage mining for example to find out the most visited web pages in the last 3 months in some website. It can also be applied to a transaction dataset in a physical store or business to find out the most frequently bought products or used services in the last time period. Any other problem that needs to study the behavior of some items with respect to time can be a good application field.

Chapter 7

Synthetic Temporal Dataset Generation

The problem of finding a suitable dataset to test different data mining algorithms and techniques and specifically association rule mining for market basket Analysis is a big challenge. A lot of dataset generators have been implemented in order to overcome this problem. *ARtool* is a tool that generates synthetic datasets and runs association rule mining for market basket analysis. But the lack of datasets that include time stamps of the transactions to facilitate the analysis of market basket data taking into account temporal aspects is notable. In this chapter, we present the *TARtool*. The *TARtool* is a data mining and generation tool based on the *ARtool* [10]. *TARtool* is able to generate datasets with time stamps for both retail and e-commerce environments taking into account general customer buying habits in such environments. We implemented the generator to produce datasets with different format to ease the process of mining such datasets in other data mining tools. An advanced GUI is also provided. The experimental results showed that our tool overcomes other tools in efficiency, usability, functionality, and quality of generated data.

7.1 Temporal Dataset Generation

Data mining is the process of finding interesting information from large databases. An important field of research in data mining is market basket analysis. The authors in [111] list the most challenging problems in data mining research. One of these problems is mining sequence data and time series data. Temporal data mining [106] is the process of mining databases that have time granularities. The goal is to find patterns which repeat over time and their periods, or finding the repeating patterns of a time-related database as well as the interval which corresponds to the pattern period [107]. The work presented in [108] studies the problem of association rules that exist in certain time intervals and thus display regular cyclic variations over time.

Most of association rule mining algorithms designed for market basket analysis have their focus on finding strong relationships between different items in the market basket data. Recently, researchers aim not only at finding relationships between different items in the market basket, but also at finding an answer to important questions with temporal dimensions such as:

- Are there any relationships between items and specific days of the week?
- Which temporal dependencies exist between distinct items?
- Which seasonal distances exist between the purchase of the same item?
- How does the sales behavior of a set of products change over time?

Because of the increasing importance of temporal data mining as an interesting field of research, researchers look for datasets that have temporal attributes to

test, develop, or improve their algorithms for temporal data mining. Those algorithms may also be of interest in many other topics of research that imply a temporal aspect such as weather data or stock market data. Unfortunately, the availability of temporal data for downloading from the web is very low, especially when we talk about temporal market basket data. This unavailability of suitable datasets leads to the conclusion, that we need to generate market basket datasets with time stamps to test and develop data mining algorithms taking into account temporal granularities.

Since market basket analysis is an important tool for improved marketing, many software solutions are available for this purpose. Business tools like *Microsoft SQL Server 2005* [112] or *IBM DB2 Data Warehouse* [113] focus on the analysis of relational business data. Software tools which are available for free download like *ARMiner* [114], *ARtool* [115], or *WEKA* [20], are more dedicated to research. They do not only analyze data, but also give additional information on the effectiveness of algorithms performed. So, in order to generate data to be used by those tools, we have to investigate which kinds of datasets can be generated.

7.2 Datasets for Association Rule Mining

A normal transaction consists of a transaction-id and a list of items in every row or sentence. Sometimes, the items are represented as boolean values 0 if the item is not bought, or 1 if the item is bought. But the commonly used format for market basket data is that of numeric values for items without any other information:

```
1 3 5 9 11 20 31 45 49
2 3 5 8 10 13 17
3 7 11 12 15 20 43...
```

This format has to be converted in order to be used by *ARMiner* and *ARtool*, since those tools can only evaluate binary data. *ARMiner* and *ARtool* have a special converter for that purpose which have to be performed before analyzing the data. *WEKA* needs a special ASCII-Data format (*.arff) for data analysis containing information about the attributes and a boolean representation of the items. Since there is no unique format for input-data, it is impossible to evaluate the same dataset in one format with different tools. In this chapter, we present a dataset generator that is able to generate datasets that are readable by *ARMiner, ARtool, WEKA,* and other data mining tools. Additionally, the generator has the ability to produce large market basket datasets with time stamps to simulate transactions in both retail and e-commerce environments.

7.3 Real World Versus Synthetic Datasets

In the beginning of data mining research, generated data was mostly used to test or develop algorithms for market basket analysis. Meanwhile, there are both generated and real world datasets available for download which can be found in data mining related websites. But both types of data suffer from different drawbacks. Real world datasets can be influenced or made unusable by:

- Incomplete or missing attributes.
- Seasonal influences like religious occasions.
- Marketing for special products.

- Good or bad weather.
- Location, e.g. in or outside the town.

Therefore, some researchers prefer generated datasets. But they also have some drawbacks such as:

- The quality of the data depends on the generator used.
- They do not really reflect customers purchasing habits.
- Algorithms performances differ when performed on generated data.

The study in [116] compares five well-known association rule algorithms using three real-world datasets and an artificial dataset from IBM Almaden. The authors showed that algorithms *Apriori, FP-Growth, Closet, Charm,* and *Magnum-Opus* do not have the same effectiveness on generated data as they have on real world data. The authors in [117] showed visually the differences between simulated data and supermarket data. They found that support and confidence of itemsets with a small support are higher in supermarket data than in generated data.

7.4 Dataset Generators and Software Solutions

There are many software solutions for data mining, and specifically market basket analysis. The following tools belong to the well-known data mining and generation software solutions.

7.4.1 The IBM Generator

In the literature, a lot of researchers use the *IBM generator* provided by the Almaden Research Center to run their experiments and test their algorithms. The generator is described in [28], and it is free for download [1]. But this generator is no longer provided by IBM, and other implementations written in C/C++, which are available for download, produce compile errors when executed.

7.4.2 The DatGen Generator

The *DatGen* generator [118] is more suitable for the purpose of classifications than association rule mining. It provides a dataset with tuples and certain constraints that can be declared. It can neither produce time stamps nor itemsets and does not meet the needs of market basket dataset.

7.4.3 An E-Commerce Generator

The dataset generator developed by Groblschegg [119] produces datasets for an e-commerce market basket. It depends on Ehrenberg's *Repeat-Buying-Theory* [120], which refers to seasonal customer habits. Ehrenberg used for his studies the so-called *consumer panels* to derive information on the intervals consumer used to purchase certain products. The generator of Groblschegg uses these intervals to generate market basket data by producing purchasing time stamps for each product and customer on a time axis. This time axis is cut by the intervals into transactions for every customer. The generator produces time stamps for the transactions as decimal numbers between 0 and 1, which are not really suitable

[1] `http://www.almaden.ibm.com/cs/quest/syndata.html`

for temporal data mining, because they cannot be mapped to attributes like specific day of a week or hour of a day.

7.4.4 ARMiner

ARMiner [114] is an open source Java tool for data mining from the University of Boston, Massachusetts. As mentioned before, it can analyze binary sequential datasets by performing different algorithms. It finds frequent itemsets and association rules depending on the specified *minsup* and *minconf* values. Effectiveness of algorithms can be evaluated by execution time and required passes over the dataset. Datasets and algorithms can be added dynamically. Furthermore, there is a test data generator available. The software is executed in the command line. There is no GUI.

7.4.5 The ARtool Generator

ARtool [115] is an enhancement of *ARMiner*, with nearly the same capabilities. It has a GUI. A few functions can only be executed in the command line. It has some more algorithms implemented, but it can only analyze the same binary datasets as *ARMiner*. *ARtool* has also a test data generator. The *ARtool* generator is a Java implementation of the *IBM generator*. It uses the parameters and variables described in [28] to generate market basket data. The parameters, which are entered in a panel of the GUI (see Figure 7.1), are as follows:

- Name and description of the dataset produced.
- Number of items.
- Number of transactions.

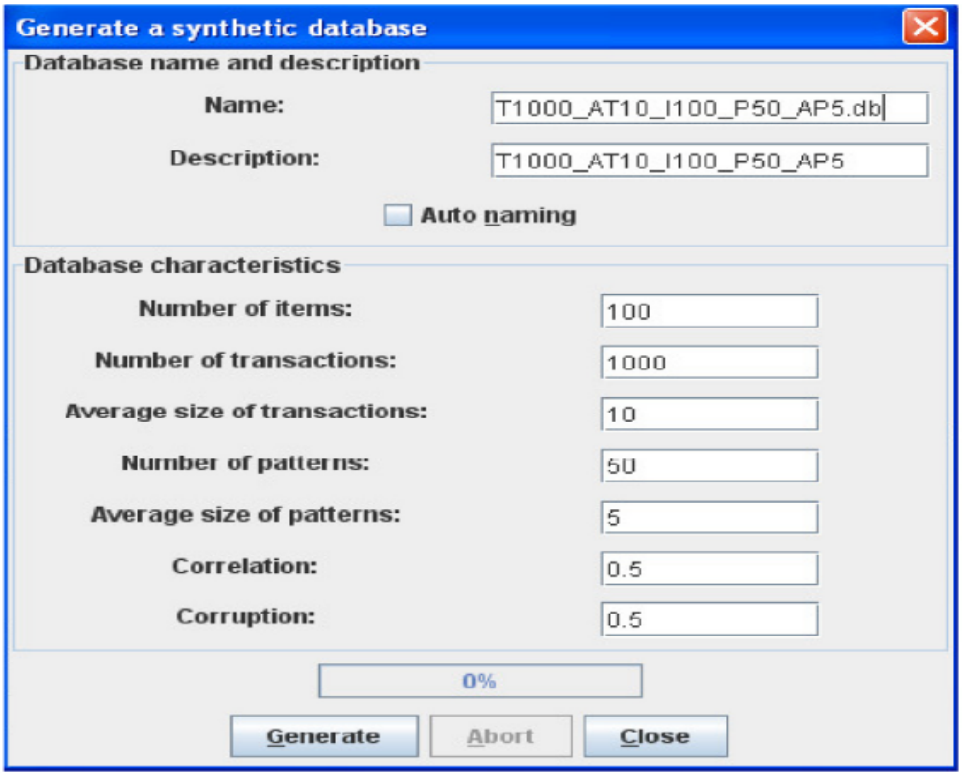


Figure 7.1: The ARtool GUI

- Average size of transactions.
- Number of patterns.
- Average size of pattern.
- Correlation level.
- Corruption level.

Defaults are provided and a progress bar shows the generation process. The output is, as mentioned, a sequential binary dataset with a header containing dataset name, description, the generated item names $C_1, ..., C_n$, the number of

transactions and items, and consecutively the transactions. To read this format, it has to be converted by a conversion tool provided by *ARtool*. The conversion tool can only be executed in the command line, not in the GUI.

7.4.6 WEKA

WEKA [20] is a tool from the University of Waikato, New Zealand, and is much more comprehensive than *ARMiner* or *ARtool*. It is designed for evaluating special ASCII-Datasets and relational data. It gives information on the effectiveness of algorithms and can visualize results. In general, it has much more functionality than *ARMiner* or *ARtool*, but it has no dataset generator available. From previous, we can say that there is no tool designed for the purpose of temporal data mining. Furthermore, there is no algorithm in any tool on-hand, that can generate dataset with time stamp functionality. We need a dataset generator that is able to generate temporal market basket data to ease the process of market basket analysis.

7.5 Enhancements for ARtool

As a result, we can say that *ARtool* synthetic dataset generator is the mostly referred to and used generator. *TARtool* is an enhancement to the *ARtool*. *TARtool* includes the following enhancements:

- Generate datasets for retail and e-commerce environments.
- Generate a time stamp for each transaction.
- Generate dataset that are readable by *ARtool, WEKA*, and other data mining tools.

- The possibility to mine those temporal datasets with algorithms provided in both tools and other data mining tools.

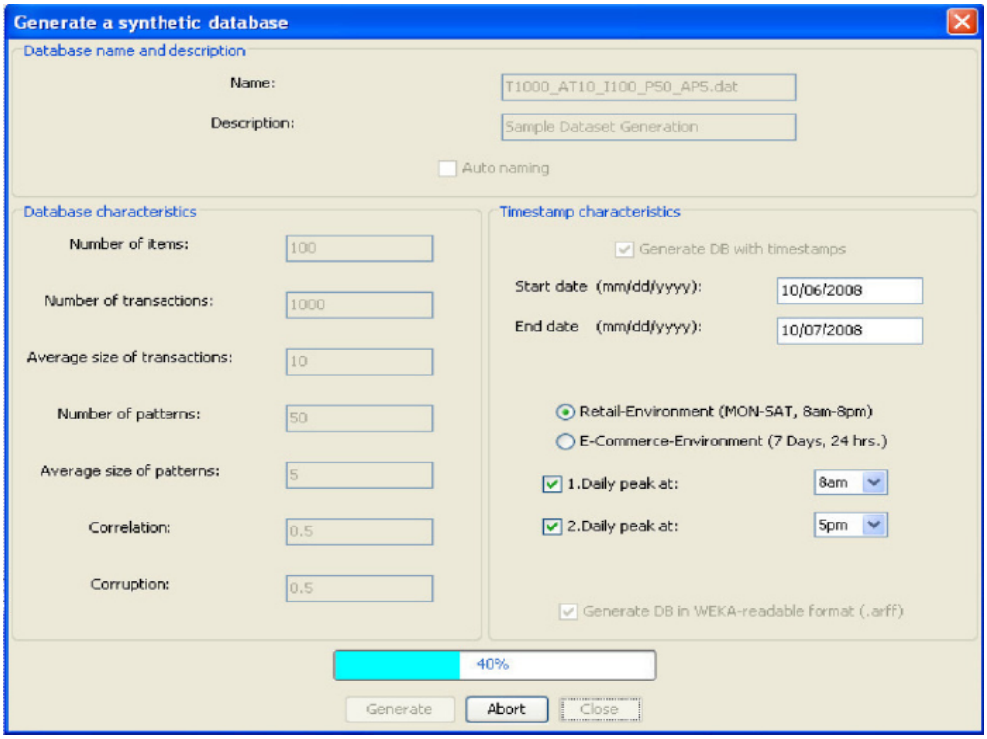


Figure 7.2: The TARtool GUI

All these features can be entered as parameters in the GUI. An example is provided in Figure 7.2. New parameters to enter are:

- Check box for dataset generation with time stamps.
- Start date and end date for the time stamps of the generated dataset.
- Buttons to choose between dataset generation for retail or e-commerce environment.
- Up to two daily peaks per day with more transactions in a specific time.

- Check box for generation of additional datasets with same values readable by *WEKA* and some other data mining tools (.arff format).

The output provided when generating a dataset with time stamp has ASCII-format with the dataset extension ".dat". An example output file looks like this:

```
1193987598500 (Fri Feb 01 08:13:18 CET 2008) 2 7 8 9 10
1193988231400 (Fri Feb 01 08:24:51 CET 2008) 4 5 8 9
1193988790000 (Fri Feb 01 08:29:10 CET 2008) 2 4 7 8 9
1193989366400 (Fri Feb 01 08:42:46 CET 2008) 4 5 6 7 8 9 10
1193989925900 (Fri Feb 01 08:52:05 CET 2008) 4 5 6 7 8 9
1193990696300 (Fri Feb 01 09:04:56 CET 2008) 2 4 5 6 7 8 9 10
```

This ASCII-file can be selected in *TARtool* as a database to be mined. The output consists of a time stamp as a 4-byte long value, containing the seconds since 1/1/1970 (the first column in the sample dataset above). This value is provided by the Java-Classes *Date* and *Gregorian Calendar* and can therefore easily be implemented in other software to handle the time stamp for analysis. Additionally, there is a readable version of the time stamp provided to see what time stamps have been generated, followed by the numerical product lists, which are generated according to the algorithm described by [28].

7.6 Time Stamp Generation

The time stamp generation depends on the *Start date* and *End date* provided in the chosen environment (i.e. retail or e-commerce). When retail environment is selected, the time stamps will be generated weekly from Monday to Saturday from 8am to 8pm. Sundays and holidays are omitted. On the other hand, If e-commerce is selected, time stamp generation will be consecutive 7 days a week,

24 hours a day. In a supermarket, there are no uniformly distributed purchases, but there are hours with a big sales volume, and hours with less. To simulate this distribution, an algorithm was implemented to achieve a rather realistic sales volume over all hours of the day. Based on the *Retail Market Basket Data Set* provided by [121] as well as the work in [122] and [123], there exists a weekend-peak of purchases from Thursday to Saturday with a sales volume of about a factor of 1.5 more than that of Monday through Wednesday.

E-commerce transactions are not comparable to classical supermarket data, because e-commerce market baskets contain normally only few purchases, for instance, items for entertainment, or technical purchases. According to [123], buyers do these purchases mainly in the after-office-hours or at weekends from Friday to Sunday. So the weekend peak in e-commerce transactions should be from Friday to Sunday.

Because daily peaks in supermarkets and e-commerce vary depending on many factors, daily peaks are not generated automatically but can be determined by parameters. If provided, they will also be rated with a factor of 1.5.

According to [122], for both environments, a reduced sales volume at night hours and daily start and end hours are generated. All these factors for sales volume rating are available as variables at the beginning of the *TARtool* and can be accessed and maintained easily if necessary. The distribution of the transactions over the chosen interval of time is computed with a poisson distribution and varied with a random factor. The poisson distribution is most commonly used to model the number of random occurrences of some phenomenon in a specified unit of space or time [124]. Items in the first itemset are chosen randomly. To model the phenomenon that large itemsets have common items, a fraction of items in

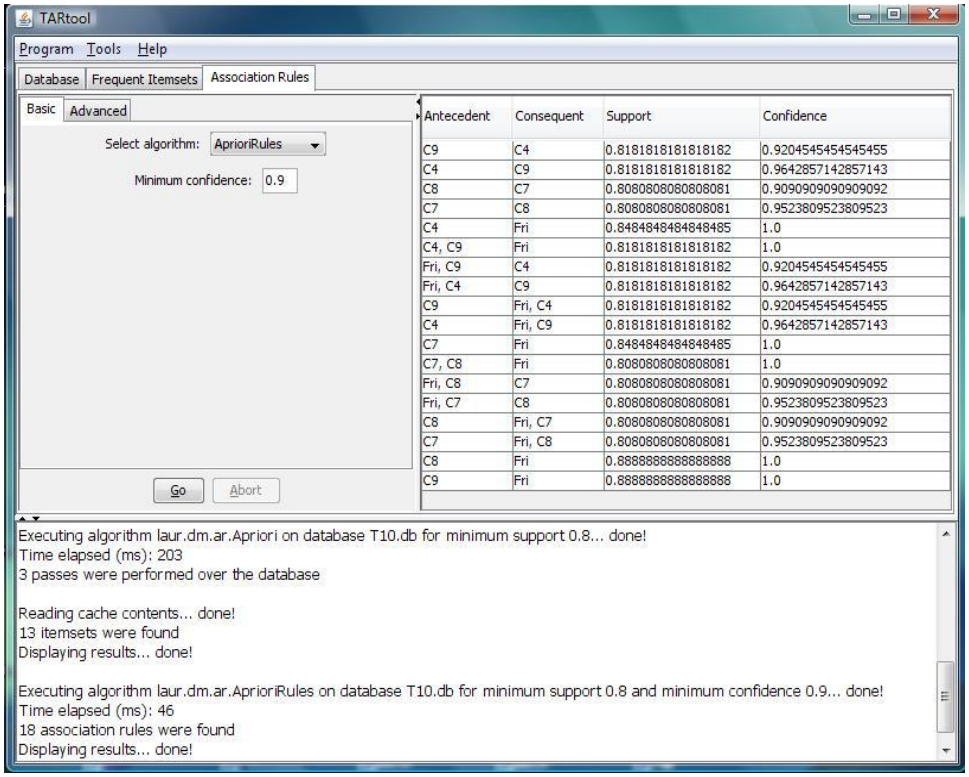


Figure 7.3: An Example of Generated Association Rules

subsequent itemsets are chosen from the previous itemset. Each itemset in the set of all large itemsets has a weight equals to the probability to pick this itemset. To model the phenomenon that all the items in large itemsets are not bought together, each itemset in the set of all large itemsets is assigned a corruption mean to indicate how much the large itemset will be corrupted before being used. Then, itemsets are assigned to the transactions. If the large itemset does not fit in the transaction, then 50% of the cases of the itemset are put in the transaction and the rest is kept for the next transaction. All generated transactions are saved in .db, .dat and/or .arff format.

The generated ASCII-file can be selected in *TARtool* for analysis like the binary

datasets. To evaluate the time stamps, the attributes week day and hour of the day from the time stamp are chosen and given as items to the association rule algorithms. An example of generated association rules can be seen in Figure 7.3. The following pseudo code summarizes the time stamped dataset generation process:

```
Variables:
- D: Number of transactions to generate
- L: Number of large itemsets to be used as
     patterns to  generate transactions
- SDate: Start-date for time stamp generation
- EDate: End-date for time stamp generation
- H: Number of days, for which transactions are to be generated
- S: Number of Sundays and holidays
- ENVMNT: True, if retail environment is selected,
          False, if E-Commerce environment is selected
- P1: First peak
- P2: Second peak
- ARFF: True, if an arff-file is to be generated
- M: Difference between the time stamps of two transactions
- V: Random Variable up to 30% of M

Begin (Dataset Generation)
1. Compute  transactions distribution
   1.1 Compute H = EDate - SDate
   1.2  If ENVMNT = True Then
              H = H - S
        Else  H = H
   1.3 Compute number of transactions per day and per hour in H
   1.4 Compute M;
       for each day and hour in H, recompute number of transactions
        1.4.1 In weekends with factor 1.5
        1.4.2 In early, late, and night hours with factor 0.5
```

```
        1.4.3 In P1 and/or P2 with factor 1.5
2. Generate Transactions
   2.1 Generate the time stamp for the next transaction
        2.1.1 Determine time stamp of the next transaction
        2.1.2 Compute V
        2.1.3 To make the time stamps to be differently
              distributed  If M is odd  Then
                                   M = M - V/M,
                           Else  M = M + V/M
   2.2 Generate the itemsets for the next transaction
        2.2.1 Determine the size of the next transaction
        2.2.2 Set the number of large itemsets to L
        2.2.3 Items in the first itemset are chosen randomly
        2.2.4 Pick itemsets
        2.2.5 Assign itemsets to the transaction
   2.3 Write transactions to output file(s): .db or .dat and
             .arff if ARFF is True
End (Dataset Generation)
```

7.7 Evaluation

To evaluate and test the efficiency of our tool, we generated a .dat-file and two related *WEKA*-readable .arff-files. They have the *WEKA*-special .arff-Format with definitions for the attributes and boolean values for the items. The first *WEKA*-readable dataset is named *.arff. It contains boolean values 0 and 1 for each item of the related .dat-file. The second file is named *q.arff and contains the values ? and 1 for the items. Because *WEKA* will find a lot of association rules within the first kind of dataset, the second one is generated to omit rules like: $IF\ A\ AND\ NOT\ B \Rightarrow C$

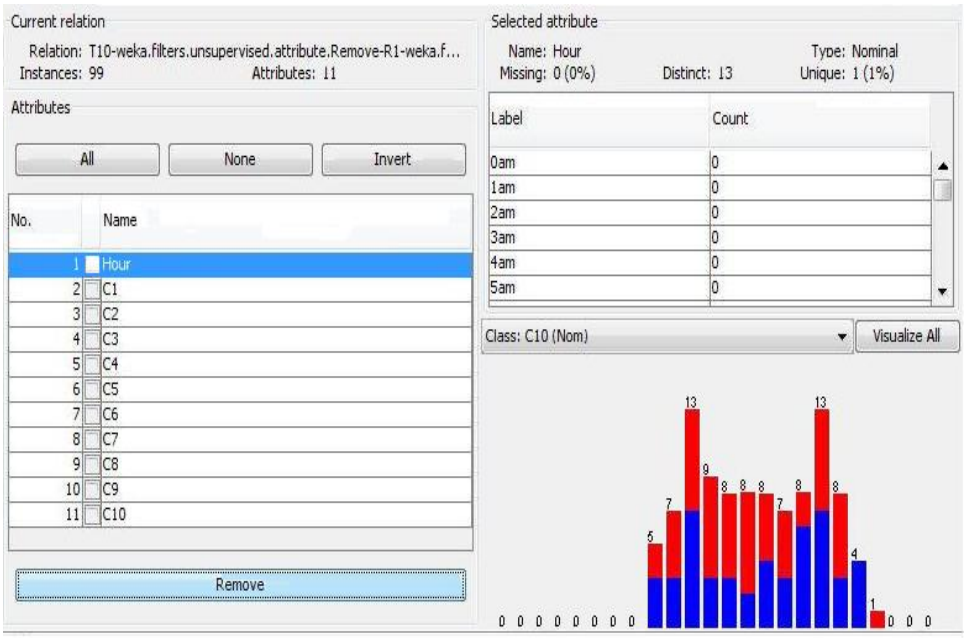
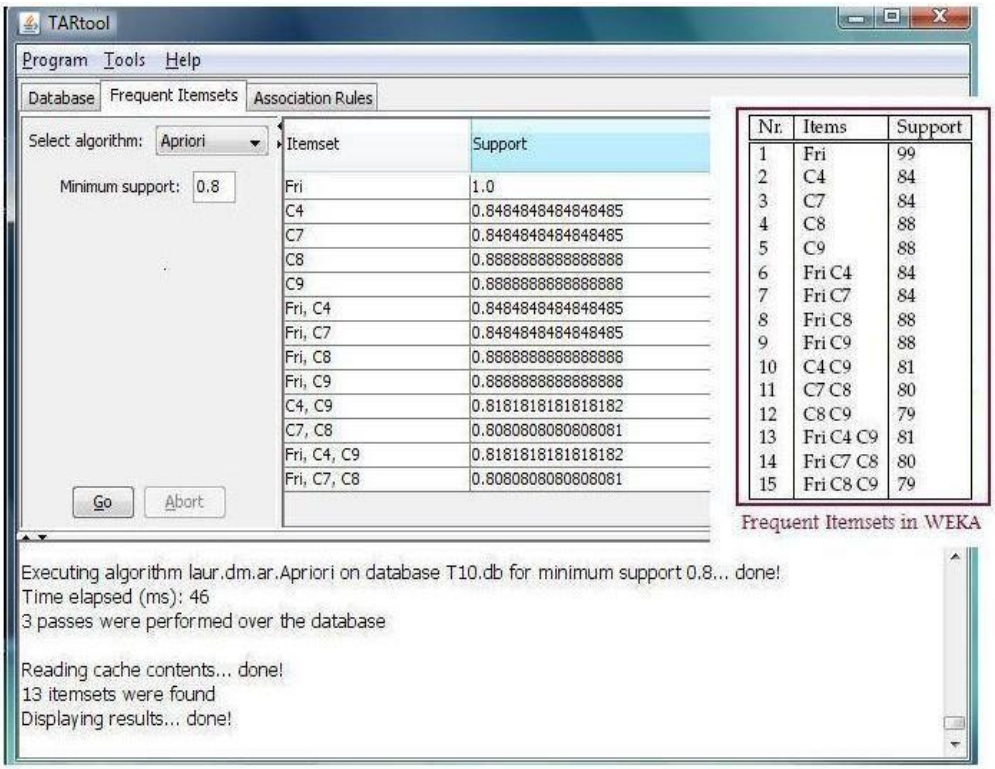


Figure 7.4: Opening of an *.arff file in WEKA

Figure 7.4 shows opening the *q.arff file in *WEKA*. The *q.arff file was generated for a retail environment with two peaks per day. The distribution of the number of transactions per hour of the day can be seen in the visualization on the right-hand side. The first peak is at 10 am and the second peak is at 5 pm. Those two peaks have been chosen in those specific time intervals in order to simulate real retail environment which, normally, have such peaks in those time points according to [123]. Then, the *q.arff-file is mined in *WEKA* and the related *.dat-file is mined in *TARtool* using the *Apriori* algorithm, which is implemented in both tools. An example of the comparison of the frequent itemsets is given in Figure 7.5. The comparison shows a very high degree of similarity of frequent itemsets under both *TARtool* and *WEKA* when mining the same generated dataset. The amount of time and space for generating binary datasets



Nr.	Items	Support
1	Fri	99
2	C4	84
3	C7	84
4	C8	88
5	C9	88
6	Fri C4	84
7	Fri C7	84
8	Fri C8	88
9	Fri C9	88
10	C4 C9	81
11	C7 C8	80
12	C8 C9	79
13	Fri C4 C9	81
14	Fri C7 C8	80
15	Fri C8 C9	79

Frequent Itemsets in WEKA

Figure 7.5: A Comparison of Frequent Itemsets in TARtool and WEKA for the Same Generated Dataset

and ASCII-files in the enhanced generator depends extensively on the parameters and the output chosen. The generation of binary datasets, i.e. those datasets without time stamps, needs much more time but less space than the ASCII-files. The generation of the *.dat-file is 10 times faster than producing the .db-file but needs many twice as much space. Table 7.1 presents some costs of generation for datasets with 100 items executed on a desktop with Pentium D Dual Core 2×1.6 GHz processor, 2048 MB RAM under Windows Vista. The generation of the binary file with 10,000,000 transactions was not executed due to the large amount of time it would have needed. But it was generated as a .dat-file.

Table 7.1: Generation Costs for Binary and ASCII Files

# of Transactions	.db Dataset		.dat Dataset	
	Millisec.	KB	Millisec.	KB
1,000	359	50	157	72
10,000	3,046	436	391	729
100,000	29,891	4,340	3,031	7,342
1,000,000	298,672	41,826	29,765	73,671
10,000,000	Not executed	...	331,875	742,164

A question which is still pending is how the generated datasets can simulate real life datasets. Even though our dataset generator can generate temporal datasets that are suitable for market basket analysis and for testing the efficiency of different algorithms, can simulate the customer transactions in both retail and e-commerce environments, and also able to handle different data format, the quality of the generated datasets are still not high enough to simulate real life customers transactions.

Therefore, we need to adjust the items generation strategy to be differently distributed for example, in some time points the number of generated items may be more or less than those in a previous or later time period depending on some criteria that simulates some buying behavior in real life transactions such as the sales amount of some group of products in a specific time period. For example, in a dataset that represents customers transactions in a grocery store, the sales of fresh bread and milk are normally more in the morning hours and in weekends than those in afternoons. Furthermore, we can develop the process of dataset generation by building a graph in which nodes represent customers who have different relationships between each other such as friendship and neighborhood.

Through those relationships, the customers influence each other in the decision of buying some product. In that way, we will be able model the purchasing process much more realistically.

Chapter 8

Conclusion and Future Work

This chapter summarizes the main contributions of this dissertation, and identifies future directions to extend this work.

8.1 Summary

In the first part of this dissertation, we developed a new method that considers data mining during the design phase of retail websites as an effective tool that participates greatly in having well-structured retail websites. The advantages of our method is that it saves a lot of maintenance efforts needed in the future. It also overcomes the drawbacks of the web mining process. Furthermore, it makes it easy for the retail decision maker to design his retail website in a way that meets the main requirements and marketing strategies of his business which will consequently increase the overall profit of the business. When using patterns extracted using association rule mining, the experimental work showed that the average cost in regard to the number of links the user may follow during the process of searching for his target products to finish a session in the improved website prototype is reduced to 62% in comparison to the average cost of the

standard website prototype. On the other hand, our method reduced the sessions average time of the improved website to 51% of the sessions average time of the standard website when using patterns extracted by applying clustering and classifcation data mining tasks to the target dataset.

In the second part of this dissertation, we studied the problem of temporal association rule mining. This approach is beneficial in market basket analysis for both physical and online shops to study customers buying habits and product buying behavior with respect to different time periods. Through periodical mining for association rules, we introduced a new method that takes into account not only interesting association rules in the decision making process, but also rules that do not meet the minimum requirements of the interestingness measurements. Applying our method in the retail business can give the decision maker a better idea about the behavior of products with respect to different time periods. It also enables him to predict/control the buying behavior of products in the next time period which will participate greatly in the success of such business.

We also presented a new measure that defines the interestingness of frequent itemsets. The interestingness measure is based on the idea that interesting frequent itemsets are supported by many recent transactions. This method can be used either as a preprocessing step to search for frequent itemsets within a determined interval, or as an extension to the *Apriori* algorithm to prune non-interesting frequent itemsets. Finally, as an approach to solve the problem of the lack of real life or synthetic transactional datasets, we introduced the *TARtool* which is a data mining tool and a synthetic dataset generator. The *TARtool* is able to generate datasets for both retail and e-commerce environments.

8.2 Future Work

As a future work to our method of using data mining to support retail websites designers during the design phase, our method can be tested on synthetic and real life datasets. A website design method that has a built-in data mining tool can be developed in order to consider the extracted interesting patterns in the website design process.

The *TARtool* can be enhanced to be able to generate biographic information of customers such as customer gender, age, and address. Further enhancements are to make it possible to determine detailed attributes of the time stamps being generated by the *TARtool* and to build visualization tools in order to ease the process of reading and analyzing the generated/mined data. Finally, the process of dataset generation can be developed by building a graph in which nodes represent customers who have different relationships between each other such as friendship and neighborhood. Through those relationships, the customers influence each other in the decision of buying some product. In this way, the purchasing process can be modeled much more realistically.

References

[1] Asem Omari and Stefan Conrad. Web Usage Mining for Adaptive and Personalized Websites. In *Proceedings of Knowledge Discovery, Data Mining, and Machine Learning (KDML06) Workshop*, pages 342–349, Hildesheim, Germany, 2006.

[2] Tec-Ed. Assessing Web Site Usability from Server Log Files. `http://www.teced.com/PDFs/whitepap.pdf`, last called in 18.05.2008, 1999.

[3] Asem Omari and Stefan Conrad. Association Rule Mining and Website's Design Improvement. In *Proceedings of the 18th GI-Workshop on the Foundations of Databases*, pages 115–119, Wittenberg, Sachsen-Anhalt, Germany, 6- 9 June 2006.

[4] Asem Omari and Stefan Conrad. On the Usage of Data Mining to Support Website Designers to Have Better Designed Websites. In *Proceedings of Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT-ICIW'06)*, volume 0, page 171, Guadeloupe, French Carribian, 2006. IEEE Computer Society.

[5] Asem Omari, Stefan Conrad and Sadet Alcic. Designing a Well-Structured E-Shop Using Association Rule Mining. In *Proceedings of the 4th International Conference on Innovations in Information Technology*, pages 9–14, Dubai, United Arab Emirates, 18- 20 November 2007. IEEE Communication Society.

[7] Asem Omari. Data Mining for Improved Website Design and Enhanced Marketing. In Yukio Ohsawa and Katsutoshi Yada, editors, *Data Mining for Design and Marketing*, volume 5 of *Chapman Hall/CRC Data Mining and Knowledge Discovery Series*, chapter 6. Chapman Hall/CRC, First edition, To appear in November 15, 2008.

[6] Asem Omari, Mehdi Bin Lamine and Stefan Conrad. On Using Clustering And Classification During The Design Phase To Build well-structured Retail Websites. In *Proceedings of the IADIS European Conference on Data Mining (ECDM2008)*, Amsterdam, The Netherlands, 2008. To appear in July 2008.

[8] Asem Omari and Stefan Conrad. On Controlling and Prediction of Next Best Sellings through Periodical Mining for {Semi/Non}-interesting Association Rules. In *Proceedings of the 1st International Conference on Digital Communications and Computer Applications (DCCA2007)*, pages 271–275. University of Science and Technology, Irbid, Jordan, March 2007.

[9] Asem Omari, Alexander Hinneburg and Stefan Conrad. Temporal Frequent Itemset Mining. In *Proceedings of the Knowledge Discovery, Data Mining*

and Machine Learning workshop (KDML 2007), Halle, Germany, September 2007. Poster.

[10] Asem Omari, Regina Langer and Stefan Conrad. TARtool: A Temporal Dataset Generator for Market Basket Analysis. In *Proceedings of the 4th International Conference on Advanced Data Mining and Applications (ADMA 2008)*, Chengdu, China, 2008. Springer Lecture Notes in Artificial Intelligence (LNAI). To appear in August, 2008.

[11] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From Data Mining to Knowledge Discovery: An Overview. In *Advances in Knowledge Discovery and Data Mining*, pages 1–34, 1996.

[12] Jiawei Han and Micheline Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco, 2001.

[13] W. H. Inmon. *Building the Data Warehouse*. John Wiley & Sons, 1996.

[14] Ramez Elmasri and Shamkant Navathe. *Fundamentals of Database Systems*. Addison Wesley, Fifth edition, 2005.

[15] Pierre F. Baldi and Kurt Hornik. Learning in Linear Neural Networks: A Survey. *IEEE Transactions on Neural Networks*, 6(4):837–858, July 1995.

[16] Pat Langley and Herbert A. Simon. Applications of Machine Learning and Rule Induction. *Communications of the ACM*, 38(11):54–64, 1995.

[17] Agnar Aamodt and Enric Plaza. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AICom - Artificial Intelligence Communications*, 7(1):39–59, 1994.

[18] L. A. Breslow and D. W. Aha. Simplifying decision trees: a survey. *Knowledge Engineering Review*, 12(1):1–40, 1997.

[19] Tom Soukup and Ian Davidson. *Visual Data Mining: Techniques and Tools for Data Visualization and Mining*. John Wiley & Sons, first edition, 2002.

[20] Ian H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, San Francisco, 2005.

[21] Margaret H. Dunham. *Data Mining Introductory and Advanved Topics*. Prentice Hall, 2003.

[22] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. CURE: An Efficient Clustering Algorithm for Large Databases. In *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pages 73–84, New York, USA, 1998. ACM.

[23] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, pages 226–231, 1996.

[24] Mihael Ankerst, Christian Elsen, Martin Ester, and Hans-Peter Kriegel. Visual Classification: An Interactive Approach to Decision Tree Construction. In *Proceedings of the fifth ACM SIGKDD International Conference on Knowledge discovery and Data Mining*, pages 392–396, 1999.

[25] J. R. Quinlan. *C4.5: Programms for Machine Learning*. Morgan Kaufmann Publishers, 1999.

[26] L. Breiman, J. Friedman, R. Olshen and C. Stone. *CART: Classification and Regression Trees*. Wadsworth Brooks/Cole Advanced Book and Software, Pacific Grove, CA., 1984.

[27] J. Hipp, U. Guntzer, and G. Nakhaeizadeh. Algorithms for Association Rule Mining - A General Survey and Comparison. *SIGKDD Explorations*, 2(1):58–64, July 2000.

[28] Rakesh Agrawal and Ramakrishnan Srikant. Fast Algorithms for Mining Association Rules. In J. B. Bocca and M. Jarke and C. Zaniolo, editor, *Proceedings of the 20th International Conference Very Large Data Bases*, pages 487–499. Morgan Kaufmann, 1994.

[29] B. Goethals. Survey on Frequent Pattern Mining. `www.adrem.ua.ac.be/~goethals/software/survey.pdf`, last called in 18.05.2008, 2003.

[30] Y. Yin J. Han, J. Pei and R. Mao. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. *Data Mining and Knowledge Discovery*, 8(1):53–87, 2004.

[31] R. Kohavi Z. Zheng and L. Mason. Real World Performance of Association Rule Algorithms. In F. Provost and R. Srikant, editor, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 401–406, 2001.

[32] C. Borgelt. An Implementation of the FP-growth Algorithm. In *Proceedings of the Workshop Open Source Data Mining Software (OSDM'05)*, pages 1–5. ACM Press, 2005.

[33] Juan Velásquez, Hiroshi Yasuda, and Terumasa Aoki. Combining the Web Content and Usage Mining to Understand the Visitor Behavior in a Web Site. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003)*, pages 669–672. IEEE Computer Society Press, 2003.

[34] Rakesh Agrawal and Ramakrishnan Srikant. Mining Sequential Patterns. In *ICDE 1995: Proceedings of the Eleventh International Conference on Data Engineering*, pages 3–14, Washington, DC, USA, 1995. IEEE Computer Society.

[35] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu. Sequential Pattern Mining Using a Bitmap Representation. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 429–435, 2002.

[36] Ruey-Shun Chen, Ruey-Chyi Wu, and J. Y. Chen. Data Mining Application in Customer Relationship Management of Credit Card Business. In *29th Annual International Computer Software and Applications Conference COMPSAC*, pages 39–40, 2005.

[37] Brian J Read. Data Mining and Science? Knowledge Discovery in Science as Opposed to Business. In *Proceedings of the 12th ERCIM Workshop on Database Research*, Amsterdam, 1999.

[38] Frank Eichinger, Detlef D. Nauck, and Frank Klawonn. Sequence Mining for Customer Behaviour Predictions in Telecommunications. In Markus Ackermann, Carlos Soares, and Bettina Guidemann, editors, *Proceedings*

of the Workshop on Practical Data Mining at ECML/PKDD, pages 3–10, Berlin, Germany, September 2006.

[39] IEEE. *IEEE Standards Collection: Software Engineering.* IEEE Standard 610.12, 1990, IEEE, 1993.

[40] Roger S. Pressman. *Software Engineering: A Practitioner's Approach.* McGraw-Hill, New York, sixth edition, 2005.

[41] W. Scacchi. Process Models in Software Engineering. pages 993–1005. John Wiley Sons Inc., 2002.

[42] Stephen R. Schach. *Software Engineering.* McGraw-Hill, Chicago, IL, second edition, 1993.

[43] A. McDonald and R. Welland. Web Engineering in Practice. In *Proceedings of the Fourth WWW10 Workshop on Web Engineering*, pages 21–30, 2001.

[44] San Murugesan and Athula Ginige. *Web Engineering: Introduction and Perspectives, Chapter 1 in Web Engineering: Principles and Techniques*, pages 1–30. Idea Group Publishing, 2005.

[45] Janice Reynolds. *The Complete E-Commerce Book: Design, Build and Maintain a Successful Web-Based Business.* CMP Books, San Francisco, 2004.

[46] Dave Gehrke. Determinants of Successful Website Design: Relative Importance and Recommendations for Effectiveness. In *HICSS 1999: Proceedings of the Thirty-second Annual Hawaii International Conference on*

System Sciences, volume 50, page 5042, Washington, DC, USA, 1999. IEEE Computer Society.

[47] O. Troyer, W. Goedefroy, and R. Meersman. UR-WSDM: Adding User Requirements Granularity to Model Web-based Information Systems. In *Proceedings of the First Workshop on Hypermedia Development Hypertext*, Pittsburgh USA, 1998.

[48] O. Troyer and C. J. Leune. WSDM: A User Centered Design Method for Web Sites. *Computer Networks*, 30(1-7):85–94, 1998.

[49] G. L. Lohse and P. Spiller. Electronic Shopping. *Communications of the ACM*, 41(7):81–88, 1998.

[50] Raymond Kosala and Hendrik Blockeel. Web Mining Research: A Survey. *ACM SIGKDD Explorations*, 2(1):1–15, 2000.

[51] Jose Borges and Mark Levene. Data Mining of User Navigation Patterns. In *Revised Papers from the International Workshop on Web Usage Analysis and User Profiling*, pages 92–111. Lecture Notes In Computer Science, 1999.

[52] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *ACM SIGKDD Explorations*, 1(2):12–23, 2000.

[53] A. Luotonen. The Common Log File Format. `http://www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format`, last called in 18.05.2008, 1995.

[54] Phillip Hallam Baker and Brian Behlendorf. Extended Log File Format. `http://www.w3.org/TR/WD-logfile-960221.html`, last called in 18.05.2008, 1996.

[55] D. Dhyani, W. Keong, and N. Bhowmick. A Survey of Web Metrics. In *ACM Computing Surveys*, volume 34, pages 469–503, 2002.

[56] Zhong Su, Qiang Yang, Hong Jiang Zhang, Xiaowei Xu, Yu-Hen Hu, and Shaoping Ma. Correlation-Based Web Document Clustering for Adaptive Web Interface Design. *Knowledge and Information Systems*, 4(2):151–167, 2002.

[57] Martha Koutri and Sophia Daskalaki. Improving Web Site Usability Through a Clustering Approach. In *Proceedings of the 10th International Conference on Human-Computer Interaction HCI, Crete, Greece*, pages 11–19, 2003.

[58] Olga Nasraoui, H. Frigui, A. Joshi, and R. Krishnapuram. Mining Web Access Logs Using Relational Competitive Fuzzy Clustering. In *Proceedings of the Eighth International Fuzzy Systems Association Congress*, Hsinchu, Taiwan, August 1999.

[59] Martin Ester, Hans-Peter Kriegel, and Matthias Schubert. Web Site Mining a New Way To Spot Competitors, Customers and Suppliers in The World Wide Web. In *Proceedings of the 8th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD 2002)*, pages 249–258, Edmonton, CA, 2002. ACM Press.

[60] Y. Fu, M. Creado, and M. Shih. Adaptive Web Sites by Web Usage Mining. In *Proceedings of the International Conference on Internet Computing (IC 2001)*, Las Vegas NA, 2001.

[61] Steffan Baron and Myra Spiliopoulou. Monitoring the Evolution of Web Usage Patterns. In *Proceedings of the 1st European Web Mining Forum EWMF*, pages 181–200, 2003.

[62] Gui-Rong Xue, Hua-Jun Zeng, Zheng Chen, Wei-Ying Ma, and Chao-Jun Lu. Log Mining to Improve the Performance of Site Search. In Bo Huang, Tok Wang Ling, Mukesh K. Mohania, Wee Keong Ng, Ji-Rong Wen, and S. K. Gupta, editors, *Proceedings of the 3rd International Conference on Web Information Systems Engineering Workshops (WISE 2002 Workshops)*, pages 238–245, Singapore, 2002. IEEE Computer Society Press.

[63] H. Yang, S. Parthasarathy, and S. Reddy. On the Use of Constrained Associations for Web. In *Proceedings of the WEBKDD Workshop: Web Mining for Usage Patterns and User Profiles*, pages 100–118, 2002.

[64] Bamshad Mobasher, Honghua Dai, Tao Luo, and Miki Nakagawa. Effective Personalization Based on Association Rule Discovery from Web Usage Data. In *WIDM 2001: Proceedings of the 3rd international workshop on Web information and data management*, pages 9–15, New York, USA, 2001. ACM Press.

[65] Bamshad Mobasher, Robert Cooley, and Jaideep Srivastava. Automatic Personalization Based on Web Usage Mining. *Communications of the ACM*, 43(8):142–151, 2000.

[66] K.L. Wu, P.S. Yu, and A. Ballman. SpeedTracer: A Web Usage Mining and Analysis Tool. *IBM Systems Journal*, 37(1), 1998.

[67] R. Cooley, J. Srivastava, and B. Mobasher. Web Mining: Information and Pattern Discovery on the World Wide Web. In *Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 1997)*, page 558, November 1997.

[68] Alex Buchner, Maurice Mulvenna, Sarab Anand, and John G. Hughes. An Internet-enabled Knowledge Discovery Process. Technical report, MINEit Software Ltd., 1999.

[69] Osmar R. Zaïane, Man Xin, and Jiawei Han. Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs. In *ADL 1998: Proceedings of the Advances in Digital Libraries Conference*, pages 1–9, Washington, DC, USA, 1998. IEEE Computer Society.

[70] Robert Cooley. The Use of Web Structure and Content to Identify Subjectively Interesting Web Usage Patterns. *ACM Transactions on Internet Technology*, 3(2):93–116, 2003.

[71] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data Preparation for Mining World Wide Web Browsing Patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.

[72] Dimitrios Pierrakos, Georgios Paliouras, Christos Papatheodorou, and Constantine D. Spyropoulos. Web Usage Mining as a Tool for Personalization: A Survey. *User Modeling and User-Adapted Interaction*, 13(4):311–372, 2003.

[73] Magdalini Eirinaki and Michalis Vazirgiannis. Web Mining for Web Personalization. *ACM Transactions for Internet Technology*, 3(1):1–27, 2003.

[74] Lara D. Catledge and James E. Pitkow. Characterizing Browsing Strategies in the World Wide Web. *Computer Networks and ISDN Systems*, 27(6):1065–1073, 1995.

[75] Mike Perkowitz and Oren Etzioni. Adaptive Web Sites: Automatically Synthesizing Web Pages. In *Proceedings of the 15th National Conference on Artificial Intelligence and 10th Innovative Applications of Artificial Intelligence Conference (AAAI 98 / IAAI 98)*, pages 727–732. AAAI Press/The MIT Press, 1998.

[76] Ramakrishnan Srikant and Yinghui Yang. Mining Web Logs to Improve Web Site Organization. In *Proceedings of the 10th International Conference on World Wide Web*, pages 430–437, Hong Kong, 2001. ACM Press.

[77] Myra Spiliopoulou and Carsten Pohle. Data Mining for Measuring and Improving the Success of Web Sites. *Data Mining and Knowledge Discovery*, 5:85–114, 2001.

[78] Alexander Mikroyannidis and Babis Theodoulidis. Web Usage Driven Adaptation of the Semantic Web. In *Proceedings of the End User Aspects of the Semantic Web Workshop, 2nd European Semantic Web Conference (ESWC05)*, pages 137–147, Heraklion, Greece, 2005. Springer-Verlag.

[79] M. Kilfoil, A. Ghorbani, W. Xing, Z. Lei, J. Lu, J. Zhang, and X. Xu. Toward an Adaptive Web: The State of the Art and Science. In *Proceedings*

of the 1st Annual Conference on Communication Networks and Services Research (CNSR 2003), pages 119–130, Moncton, Canada, 2003.

[80] Peter Brusilovsky. Efficient Techniques for Adaptive Hypermedia. In *Intelligent Hypertext*, pages 12–30, 1997.

[81] J. Fink, A. Kobsa, and A. Nill. User-oriented Adaptivity and Adaptability in the Avanti Project. In *Proceedings of the Conference Designing for the Web: Empirical Studies*, Microsoft Campus Redmond, USA, 1996.

[82] Alan Wexelblat. An Environment for Aiding Information-browsing Tasks. In Gil., Birmingham, Cypher, and Pazzani, editors, *Proceedings of AAAI Spring Symposium on Acquisition, Learning and Demonstration: Automating Tasks for Users*, Birmingham, UK, 1996. AAAI Press.

[83] Peter Brusilovsky. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, 6:87–129, 1996.

[84] Mike Perkowitz and Oren Etzioni. Adaptive Web Sites. *Communications of the ACM*, 43(8):152–158, 2000.

[85] Bettina Berendt and Myra Spiliopoulou. Analysis of Navigational Behavior in Web Sites Integrating Multiple Information System. *VLDB Journal*, 9:56–75, 2000.

[86] SaiMing Au. A Study of Application of Web Mining for E-Commerce: Tools and Methodology. *International Journal of The Computer, The Internet and Management*, 10(3):1–14, 2002.

[87] M. Koutri, N. Avouris, and S. Daskalaki. A Survey on Web Usage Mining Techniques for Web-based Adaptive Hypermedia Systems. In *Proceedings of the Adaptable and Adaptive Hypermedia Systems*, pages 125–149. IRM Press, 2005.

[88] Charalampos Vassiliou, Dimitris Stamoulis, and D. Martakos. The Process of Personalizing Web Content: Techniques, Workflow and Evaluation. In *Proceedings of the SSGRR International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet*, 2002.

[89] Dong-Ho Kim, Vijayalakshmi Atluri, Michael Bieber, Nabil Adam, and Yelena Yesha. A Clickstream-based Collaborative Filtering Personalization Model: Towards a Better Performance. In *WIDM 2004: Proceedings of the 6th annual ACM international workshop on Web information and data management*, pages 88–95, New York, USA, 2004. ACM Press.

[90] C. Shahabi, A. M. Zarkesh, J. Adibi, and V. Shah. Knowledge Discovery from Users Web-page Navigation. In *RIDE 1997: Proceedings of the 7th International Workshop on Research Issues in Data Engineering RIDE 1997 High Performance Database Management for Large-Scale Applications*, page 20, Washington, DC, USA, 1997. IEEE Computer Society.

[91] Hiroshi Ishikawa, Manabu Ohta, Shohei Yokoyama, Junya Nakayama, and Kaoru Katayama. On the Effectiveness of Web Usage Mining for Page Recommendation and Restructuring. In Akmal B. Chaudhri, Mario Jeckle, Erhard Rahm, and Rainer Unland, editors, *Proceedings of Web, Web-Services,*

and Database Systems -NODe Web and Database-Related Workshops, pages 253–267. Springer-Verlag, 2002.

[92] Dimitrios Pierrakos, Geogios Paliouras, Christos Papatheodouro, and Constantine D. Spyropoulos. KOINOTITES: A Web Usage Mining Tool for Personalization. In *Proceedings of the Panhellenic Conference on Human Computer Interaction*, 2001.

[93] Massimiliano Albanese, Antonio Picariello, Carlo Sansone, and Lucio Sansone. A Web Personalization System Based on Web Usage Mining Techniques. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers and posters*, pages 288–289, New York, USA, 2004. ACM Press.

[94] Nicholas Kushmerick, James McKee, and Fergus Toolan. Towards Zero-Input Personalization: Referrer-Based Page Prediction. *Lecture Notes in Computer Science*, 1892:133–143, 2000.

[95] Shian-Hua Lin and Jan-Ming Ho. Discovering Informative Content Blocks from Web Documents. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2002)*, pages 588–593. ACM Press, 2002.

[96] Bing Liu, Yiming Ma, and Philip S. Yu. Discovering Unexpected Information from Your Competitors' Web Sites. In *Proceedings of 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (KDD 2001)*, pages 144–153. ACM Press, 2001.

[97] Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. Mining Product Reputations on the Web. In *KDD 2002: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 341–349, New York, USA, 2002. ACM Press.

[98] Brian D. Davison. Predicting Web Actions from HTML Content. In *Proceedings of the Thirteenth ACM Conference on Hypertext and Hypermedia*, pages 159–168, College Park, MD, June 2002.

[99] Bing Liu, Robert Grossman, and Yanhong Zhai. Mining Data Records in Web Pages. In Lise Getoor, Ted E. Senator, Pedro Domingos, and Christos Faloutsos, editors, *Proceedings of 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2003)*, pages 601–606. ACM Press, 2003.

[100] William W. Cohen. Learning and Discovering Structure in Web Pages. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 26(1):3–10, 2003.

[101] Vassil Gedov, Carsten Stolz, Ralph Neuneier, Michal Skubacz, and Dietmar Seipel. Matching Web Site Structure and Content. In Stuart I. Feldman, Mike Uretsky, Marc Najork, and Craig E. Wills, editors, *Proceedings of the 13th International Conference on World Wide Web (WWW 2004) Alternate Track Papers and Posters*, pages 286–287. ACM Press, 2004.

[102] Ding Cai, Xiaofei He, Ji-Rong Wen, and Wei-Ying Ma. Block-Level Link Analysis. In Mark Sanderson, Kalervo Järvelin, James Allan, and Peter

Bruza, editors, *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR04)*, pages 440–447. ACM Press, 2004.

[103] Elaine Chou. Redesigning a Large and Complex Website: How to Begin, and a Method for Success. In *SIGUCCS 2002: Proceedings of the 30th annual ACM SIGUCCS conference on User services*, pages 22–28, New York, USA, 2002. ACM Press.

[104] F. Comite, R. Gilleron, and M. Tommasi. Learning Multi-label Alternating Decision Trees and Applications. In Gilles Bisson, editor, *Proceedings of CAp01 : Conference en Apprentissage Automatique*, pages 195–210, 2001.

[105] Yoav Freund and Llew Mason. The Alternating Decision Tree Learning Algorithm. In *Proceedings of the 16th International Conference on Machine Learning*, pages 124–133. Morgan Kaufmann, San Francisco, CA, 1999.

[106] Claudia Antunes and Arlindo L. Oliveira. Temporal Data Mining: An Overview. In *Proceedings of the Workshop on Temporal Data Mining, of Knowledge Discovery and Data Mining (KDD01)*, San Francisco, USA, 2001.

[107] Weiqiang Lin, Mehmet Orgun, and Graham J. Williams. An Overview of Temporal Data Mining. In *Proceedings of the 1st Australian Data Mining Workshop*, pages 83–90. University of Technology, Sydney, 2002.

[108] Banu Ozden, Sridhar Ramaswamy, and Abraham Silberschatz. Cyclic Association Rules. In *ICDE '98: Proceedings of the Fourteenth International*

Conference on Data Engineering, pages 412–421, Washington, DC, USA, 1998. IEEE Computer Society.

[109] Ding-An Chiang, Shao-Lun Lee, Chun-Chi Chen, and Ming-Hua Wang. Mining Interval Sequential Patterns. *International Journal of Intelligent Systems*, 20(3):359–373, 2005.

[110] Kaidi Zhao and Bing Liu. Visual Analysis of The Behavior of Discovered Rules. In *Workshop Notes in ACM SIGKDD-2001 Workshop on Visual Data Mining*, San Francisco, CA, 2001.

[111] Qiang Yang and Xindong Wu. 10 Challenging Problems in Data Mining Research. *International Journal of Information Technology and Decision Making*, 5(4):597–604, 2006.

[112] Microsoft. SQL Server 2005 Data Mining Tutorial. `http://msdn2.microsoft.com/en-us/library/ms167167.aspx`, last called in 18.05.2008.

[113] IBM. DB2-Business-Intelligence: Overview of the Data Mining Features.

[114] Laurentiu Cristofor. ARMiner Project. University of Massachusetts, Boston. `http://www.cs.umb.edu/laur/ARMiner/`, last called in 18.05.2008.

[115] Laurentiu Cristofor. ARtool Project. University of Massachusetts, Boston. `http://www.cs.umb.edu/laur/ARtool/`, last called in 18.05.2008.

[116] Zijian Zheng, Ron Kohavi, and Llew Mason. Real World Performance of Association Rule Algorithms. In Foster Provost and Ramakrishnan Srikant,

editors, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 401–406, 2001.

[117] Michael Hahsler and Kurt Hornik. New Probabilistic Interest Measures for Association Rules. In *Intelligent Data Analysis*, volume 11, pages 437–455, 2007.

[118] Gabor Melli. Dataset Generator. `http://www.datasetgenerator.com`, last called in 18.05.2008.

[119] Martin Groblschegg. Developing a Testdata Generator for Market Basket Analysis for E-commerce Applications. Vienna University of Economics and Business Administration, 2003.

[120] Andrew Ehrenberg. Repeat-Buying: Facts, Theory and Applications. London, 1988. Charles Griffin & Company Ltd.

[121] Tom Brijs, Gilbert Swinnen, Koen Vanhoof, and Geert Wets. Using Association Rules for Product Assortment Decisions: A Case Study. In *Proceedings of Knowledge Discovery and Data Mining*, pages 254–260, 1999.

[122] Humberto T. Marques Neto, Jussara M. Almeida, Leonardo C. D. Rocha, Wagner Meira, Pedro H. C. Guerra, and Virgilio A.F. Almeida. A Characterization of Broadband User Behaviour and Their E-Business Activities. *SIGMETRICS Performance Evaluation Review, Special Issue: E-Commerce*, 32:3–13, 2004.

[123] Udaykiran Vallamsetty, Krishna Kant, and Prasant Mohapatra. Characterization of E-commerce Traffic. In *Proceedings of the International Workshop*

on Advanced Issues of E-Commerce and Web Based Information Systems, pages 137–147, Los Alamitos, California, 2002. IEEE Computer Society.

[124] Granino A. Korn and Theresa M. Korn. *Mathematical Handbook for Scientists and Engineers: Definitions, Theorems, and Formulas for Reference and Review*. Dover Publications, 2000.