T.C.

ALTINBAS UNIVERSITY

Information Technology

DESIGN AND IMPLEMENTATION OF A PROTOTYPE DATA MINING AGENT SYSTEM

Ali Abdulhussein Mohammed

Master Thesis

Supervisor: Dr. Sefer Kurnaz

Istanbul, 2019

DESIGN AND IMPLEMENTATION OF A PROTOTYPE DATA MINING AGENT SYSTEM

by

Ali Abdulhussein Mohammed

Information Technology

Submitted to the Graduate School of Science and Engineering

in partial fulfillment of the requirements for the degree of

Master of Science

ALTINBAŞ UNIVERSITY

2019

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of .....

Asst. Prof. Dr. Sefer KURNAZ
Supervisor

Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)

Asst. Prof. Dr. Osman N. UCAN	School of Engineering and Natural Scince, Altinbaş University	_____
Asst. Prof. Dr. Sefer KURNAZ	School of Engineering and Natural Scince, Altinbaş University	_____
Asst. Prof. Name SURNAME	School of Engineering and Natural Scince, Altinbaş University	_____

I certify that this thesis satisfies all the requirements as a thesis for the degree of .....

Asst. Prof. Dr. OĞUZ ATA
Head of Department

Approval Date of Graduate School of Science and Engineering: ____/____/____

Asst. Prof. Dr.OĞUZ BAYAT
Director

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ali Abdulhussein Mohammed

DEDICATION

First and foremost, I would like to thank Allah Almighty for giving me the knowledge, ability and opportunity to undertake this research study and to persevere and complete it satisfactorily. Heartfelt thanks goes to my father and my mother. Every success is a direct consequence of their influence in my life and their love. At the end I have to mention my family and Hussein Nariman brotherhood for their support and love.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Supervisor Dr. Sefer KURNAZ for all the knowledge and support he provided during my study for the Master Degree and throughout the work to complete this thesis and I have to mention the kindness and the support to all my friends particularly Ali yousif which did not leave me alone the whole time at the courses and while doing this thesis.

ABSTRACT

DESIGN AND IMPLEMENTATION OF A PROTOTYPE DATA MINING AGENT SYSTEM

Ali Abdulhussein Mohammed

M.Sc., Information Technologies, Altınbaş University

Supervisor. Dr. Sefer KURNAZ

Date: March 2019

Pages: 51

An analytical, descriptive and perhaps instinctive methodology regarding sourcing out ways to improve business opportunities and technical know-how is incorporated within the scope of Data Mining. Its designation relates to how an event deals with obtaining unprocessed information from vast databases by deeply searching the resourceful highland areas for further processing to qualify valuableness. Its applications are typically employed in facilitating the operations of marketing, sales and services of business though it works more with stable datasets that are kept in the business's threshold of sensitive and discreet information. The essence of its strategies in the modern world includes automated prediction of trends and behavior as well as abstracting initially unknown patterns. Grave utterances concerning the supposedly efficient data mining techniques have been made significant over the years since its discovery. There are still positive and negative remarks about these said techniques claimed to improve business functionalities. The former which may involve how increased processing speeds, lowered costs for storage and better software packages tend to make data mining more economical. And as for the latter, involving how it is deemed unethical to use the processes of data mining as it violates innocent people's privacy. Data mining can be quite an enigma for the inadequately learned individuals and firms due to its cognitive requirements that appear to be highly critical and extensive upon application.

Keywords: Data Mining Methods, Data Mining Strategies, Data Mining Process, Applications of Data Mining, Importance of Data Mining**.**

TABLE OF CONTENTS

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

LIST OF ABBREVIATIONS

ARD	:	Association Rule Database
ADA	:	Automating Data Analysis
DRFFSM1	:	Develop Random Forests with Fuzzy Similarity Measure1
DRFFSM2	:	Develop Random Forests with Fuzzy Similarity Measure2
DRFFSM3	:	Develop Random Forests with Fuzzy Similarity Measure3
DRFPC	:	Develop Random Forests with Pearson Correlation
DRFSS	:	Develop Random Forests with Simple Similarity
DRFLLS	:	Developed Random Forest and Local Least Square
FP	:	Frequency Pattern

1. INTRODUCTION

The concept of data mining (DM) is well-known in different areas such as business, engineering, communications, transport, medicine, education etc. Data can be transformed into usable knowledge as part of knowledge management initiatives using data mining techniques to increase organizations' assets. Data from various activities fields are produced and stored daily, processed, transmitted in different locations without taking into account their meanings. Managers focus their activity mainly on finding methods and techniques to organize huge data provided by transactions or other activities and to extract useful patterns, relations, associations from data etc [1].

Data mining is a technique that discovers previously unknown relationships in data. Although data mining is a valuable technology for many application domains, it has not been widely adopted by business users at large and by the database community more specifically. This can be largely attributed to the nature of the data mining tasks. Data mining is a difficult and laborious activity that requires a great deal of expertise for obtaining quality results. It is also a multidisciplinary it often requires the expertise of numerous individuals working in the selection of techniques, creation of models, and parameter tuning in order to support analytical activities [2].

Despites its high claims and expectations, DM technology requires a highly trained professional to do an iterative, multistep process of accessing and preparing data, choosing an appropriate algorithm to mine the dat analyzing the learned knowledge, and presenting nontrivial, valuable knowledge to executives or decision makers [3].

"Data mining is a difficult and laborious activity that requires a great deal of expertise for obtaining high quality results". New methods are necessary for intelligent data analysis to extract relevant information with minimum effort. Even choosing the correct data mining algorithm involves more time for the system. A solution for this problem could be an intelligent system based on agents [4].

Mohammed J. Zaki and Limsoon Wong considered automation as one of the research challenges for data mining from the perspectives of scientific and engineering applications are issues. While a data mining algorithm and its output may be readily

handled by a computer scientist, it is important to realize that the ultimate user is often not the developer. In order for a data mining tool to be directly usable by the ultimate user, issues of automation especially in the sense of ease of use must be addressed. Even for the computer scientists, the use and incorporation of prior knowledge into a data mining algorithm is often a tricky challenge; They too would appreciate if data mining algorithms can be modularized in a way that facilitate the exploitation of prior knowledge [4].

On the other hand, Agent-based systems (ABS) belong to the most vibrant and important areas of research and development to have emerged in information technology in the 1990s [5].

Agents have become increasingly popular in computing world in recent years. Some of the reasons for this popularity are their flexibility, modularity and general applicability to a wide range of problems (data filtering and analysis, information brokering, condition monitoring and alarm generation, workflow management, personal assistance, simulation and gaming). Because of the explosive development of information source available on the Internet and on the business, government, and scientific databases, it has become quite necessary for the users to utilize automated_and intelligent tools to extract knowledge from them. Agents can help making computer systems easier to use, enable finding and filtering information, customizing views of information and automating work. An Intelligent agent is software that assists people and acts on their behalf. Intelligent agents work by allowing people to delegate work that they could have done to the agent software [6].

Knowledge discovery process can be assisted by agents in order to increase the quality of knowledge and to simplify the main processes of identifying patterns from huge data volumes. Intelligent agents and data mining share the same objectives in order to assist decision making process. Data mining and agents can make a common front to help people in the decision making process, to elaborate decisional models and take good decision in real time.

1.1 LITERATURE SURVEY

In 1994, Winton Davies was the first to work with the agent-based data mining field. His start was the guide for others to continue, especially that he give a lot of suggestion on the integration between the two fields.

• P. A. Mitkas, A.L. Symeonidis, D. Kehagias and I. Athanasiadis, 2003, developed Agent Academy, a software platform for the design, creation, and deployment of multi-agent systems, which combines the power of knowledge discovery algorithms with the versatility of agents. Using this platform, they show how agents, equipped with a data-driven inference engine, can be dynamically and continuously trained. They also discuss a few prototype multi-agent systems developed with Agent Academy [7].

• M. Campos, Peter J. Stengard, and Boriana L. Milenova, 2005, proposed a new approach to the design of data mining applications targeted at database and BI user groups. This approach uses a data-centric focus and automated methodologies to make data mining accessible to non-experts. The automated methodologies are exposed through high-level interfaces [8].

• Arti Rana, Ashish Jolly, and Priyanka Pawar, 2004, used two studies to show how data mining can make the difference during the knowledge management process. First, we described how data mining was used as part of knowledge management. Second, Intelligent agents can be used as data mining tool to extract the meaningful information from the raw data [27].

• Nittaya Kerdprasop and Kittisak Kerdprasop, 2007, described a framework of an intelligent and complete data mining system called SUT-Miner. The system is comprised of a full complement of major DM algorithms, pre-DM and post-DM functionalities [17].

• Irina Tudor and Liviu Ionita, 2009, considered an example of "data mining agents", outlining their major involvement in the complex process of knowledge management in academic environment [9].

• Serge Parshutin and Arkady Borisov, 2009, presents the Agents Based Data Mining and Decision Support system in the field of production planning, meant for supporting a production manager in

his/her production planning decisions. The developed system is based on the analysis of historical demand for products and on the information about transitions between phases in life cycles of those products. The architecture of the developed system is presented as also an analysis of testing on the real-world data results is given [10].

1.2 THESIS OBJECTIVE

The main objective of this work is to design and implement an automated data mining system using a software agent. The role of the agent can be summarized in the following steps:

- Search for the presence of any database files in the environment and load it to the program.
- Apply the data mining system stages like preprocessing deciding the suitable data mining task, executing the algorithm and obtaining results.
- Store the results in a knowledgebase file.

2. DATA MINING SYSTEMS

The problem of understanding characteristics of data has attracted keen interest of scientists from early years of computer science but it increases in the past two decades since the dramatic increase in the amount of data being stored in electronic format (Figure 2.1). This accumulation of data has taken place at an aggressive rate. It has been estimated that the amount of data in the world doubles every 20 months and the size and number of databases are increasing even faster [23].

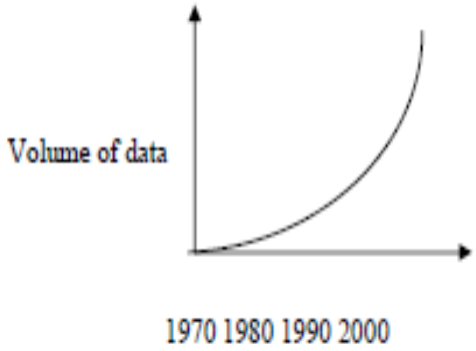


Figure 2.1: The massive increase in data

As the volume of data increases, inexorably, analyzing such data turned out to be much harder than expected, and the proportion of it that people understand decreases, alarmingly [11].

Obviously, there is a need for developing techniques and tools that assist users to analyze and automatically extract hidden knowledge [18].

which are provided by data mining.The ongoing remarkable growth in the field of data mining and knowledge discovery has been fueled by a fortunate confluence of a variety of factors [12].

• The explosive growth in data collection.
• The tremendous growth in computing power and storage capacity.
• The availability of increased access to data from web navigation and intranets.
• The development of database management systems.

2.1 DATA MINING DEFINITIONS

Numerous and different definitions are given to data mining, however they all meet at the same meanings According to the Gartner Group," Data mining is the process of discovering meaningful new correlations, patterns and trends by sifting through large amounts of data stored in repositories using pattern recognition technologies as well as statistical and mathematical techniques" [20].

In the words of Jiawei Han and Micheline Kamber "data mining refers to extracting or mining knowledge from large amounts of data" [13].

Another definition considers that "data mining is the automatic discovery of previously unknown patterns or relationships in large and complex datasets" [17].

"Data mining refers to using a variety of techniques to identify nuggets of information or decision making knowledge in bodies of data, and extracting these in such a way that they can be put to use in the areas such as decision support, prediction, forecasting and estimation" [23].

2.2 DATA MINING VERSUS KNOWLEDGE DISCOVERY

Many people treat data mining as a synonym for another popularly used term, Knowledge Discovery in Databases, or KDD. Alternatively, others view data mining as simply an essential step in the process of knowledge discovery [6].

Knowledge discovery consists of an iterative sequence of the following steps [13]

- **Selection:** selecting or segmenting data relevant to the analysis task e.g. all those people who own a car; in this way subsets of the data can be determined.
- **Preprocessing:** this is the data cleansing stage where certain information is removed which is deemed unnecessary. Also data is reconfigured to ensure a consistent format.
- **Transformation :** where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations, for instance.
- **Data mining :**an essential process where intelligent methods are applied in order to extract data patterns.

- **Interpretation and evaluation:** The patterns identified by the system are interpreted into knowledge which can be then used to support human decision-making. Also, to identify the truly interesting patterns representing knowledge based on some interestingness measures So it is obvious that data mining is a step in the knowledge discovery process. However, in industry, in media, and in the database research milieu, the term data mining is becoming more common and popular than the longer term of knowledge discovery from data Accordingly, the term data mining is used in this thesis,and the term data mining system is used for the proposed system.

Figure 2.2: Data Mining and KDD

2.3 DATA MINING SYSTEMS

Jiawei Han and Micheline Kamber explained the architecture of a typical data mining system having the following major components [14].

- Database, data warehouse, World Wide Web, or other information repository this is one or a set of databases, data warehouses, spreadsheets, or other kinds of information repositories. Data cleaning and data integration techniques may be performed on the data.

- Database or data warehouse server**:** The database or data warehouse server is responsible for fetching the relevant data, based on the user's data mining request.

- Knowledge base**:**This is the domain knowledge that is used to guide the search or evaluate the interestingness of resulting patterns.
- Data mining engine**:**This is essential to the data mining system and ideally consists of a set of functional modules for tasks such as association analysis, classification, prediction, and cluster analysis.
- Pattern evaluation module: This component typically employs interestingness measures and interacts with the data mining modules so as to focus the search toward interesting patterns.

2.4 CLASSIFICATION OF DATA MINING SYSTEMS

Data Mining systems can be categorized according to various criteria, as follows [15].

- Classification according to the kinds of databases mined**:** A data mining system can be classified according to the kinds of databases mined. Database systems can be classified according to different criteria, each of which may require its own data mining technique. Data mining systems can therefore be classified accordingly.

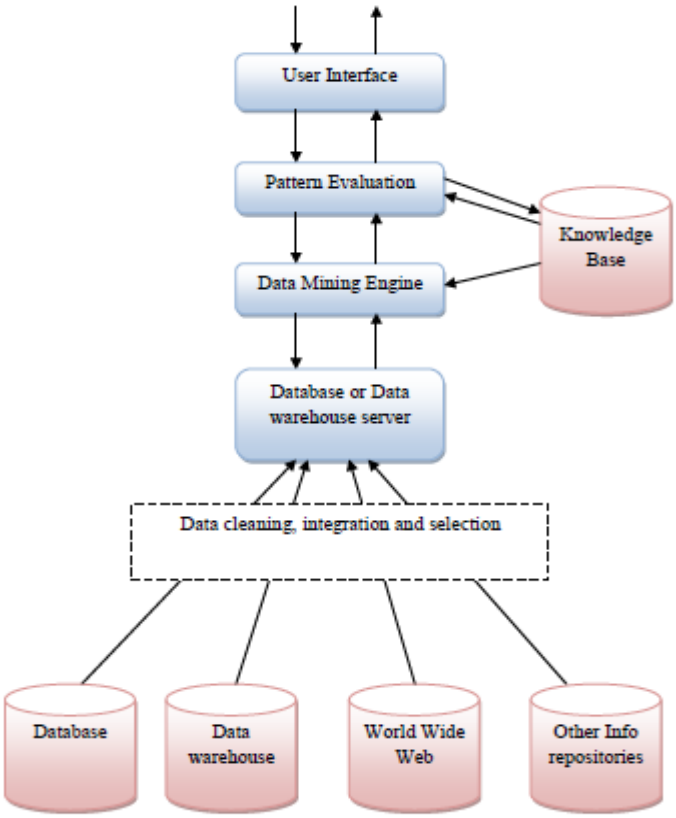


Figure 2.3: Architecture of a typical data mining system

- Classification according to the kinds of knowledge mined Data mining systems they mine, that is, based on data mining functionalities, such as association analysis, classification, prediction and clustering. A comprehensive data mining system usually provides multiple and/or integrated data mining functionalities. Moreover, data mining systems can be distinguished based on the granularity or levels of abstraction of the knowledge mined, including generalized knowledge (at a high level of abstraction) primitive-level knowledge (at a raw data level), or knowledge at multiple levels (considering several levels of abstraction). An advanced data mining system should facilitate the discovery of knowledge at multiple levels of abstraction.

- Classification according to the kinds of techniques utilized**:** Data mining systems can be categorized according to the underlying data mining techniques employed. These techniques can be described according to the degree of user interaction involved (e.g., autonomous systems, interactive exploratory systems, query-driven systems) or the methods of data analysis employed (e.g., database-oriented or data warehouse–oriented

techniques, machine learning, statistics, visualization, pattern recognition, neural networks, and so on). A sophisticated data mining system will often adopt multiple data mining techniques or work out an effective, integrated technique that combines the merits of a few individual approaches.

- Classification according to the applications adapted**:** Data mining systems can also be categorized according to the applications they adapt. For example, data mining systems may be tailored specifically for finance, telecommunications, DNA, stock markets, e-mail, and so on. Different applications often require the integration of application-specific methods. Therefore, a generic, all-purpose data mining system may not fit domain-specific mining tasks.

2.5 INPUT TO DATA MINING SYSTEMS

2.5.1 Metadata Information

Metadata is data about data which describes the content, quality, condition and other characteristics of data. It provides information that will be used by analysts in understanding the data and building the models. It plays an important role not only in the design, implementation and maintenance of data warehouses, but also in data organizing, information querying and result understanding [31]. The application of metadata can support the selection of suitable mining methods as well as of appropriate parameter values to control these methods [24].

There are many separate kinds of metadata information ranging from simple description to a more detailed one [7].

2.5.2 Kinds of Data Repositories

The data of which data mining is applied to, can be stored in different kinds of repositories and in different shapes. Some of these kinds are [40].

- Flat file database Flat files are actually the most common data source for data mining algorithms, especially at the research level. Flat files are simple data files in text or binary format with a structure known by the data mining algorithm to be applied. The data in these files can be transactions, time-series data, scientific measurements, etc…

- Relational databases**:** A relational database is a collection of tables, each of which is assigned a unique name. Each table consists of set of attributes (columns) and usually stores a large set of tuples.Each tuple represents an object identified by a unique key, and described by a set of features or attribute values. The value of an attribute for a particular instance is a measurement of the quantity to which the attribute refers. There is a broad distinction between quantities that are numeric and ones that are nominal. Numeric attributes, sometimes called continuous attributes, measure numbers either real or integer valued. Nominal attributes take on values in a prespecified finite set of possibilities and are sometimes called categorical or discrete. Most practical data mining systems accommodate just these two attribute types: numeric and nominal.
- Data Warehouses**:** A data warehouse is a repository of information collected from multiple sources, stored in a unified schema, and usually resides at a single side. Data warehouses are constructed via a process of data cleaning, data integration, data transformation and data loading.
- Advanced data and information systems**:** With the progress of database technology, various kinds of advanced data and information systems have emerged and are undergoing development to address the requirements of new applications. Some of these new kinds:

a) Object-relational databases
b) Temporal Databases, Sequence Databases, and Time-Series Databases
c) Spatial Databases and Spatiotemporal Databases
d) Text Databases and Multimedia Databases
e) Heterogeneous Databases and Legacy Databases
f) The world wide web

2.6 DATA PREPROCESSING

Data preparation is a fundamental stage of data analysis. While a lot of low-quality information is available in various data sources and on the Web, many organizations or companies are interested in how to transform the data into cleaned forms which can be used for high-profit purposes. This goal generates an urgent need for data analysis aimed at cleaning the raw data [42].

2.7 DESCRIPTIVE DATA SUMMARIZATION

Descriptive data summarization techniques provides the analytical foundation for data preprocessing, and can be used to identify data properties, and detect the noise or outliers. This can be achieved by measuring the central tendency and dispersion of the data.

a) Measuring the Central Tendency

b) Measuring the dispersion of data

Dispersion or variance is the degree to which numerical data tend to spread. Range Graphical representations, such as histograms facilitate visual inspection of the data and are thus useful for data preprocessing and mining.

2.7.1 Data cleaning

Much of the raw data contained in databases is unpreprocessed incomplete, and noisy [16].

Handling missing values is one of the important concepts in data cleaning.Many ways are used to handle such cases [16].

a) Ignore the tuple.
b) Replace all missing values with a single global constant (a selection of a global constant is highly application-dependent).
c) Replace a missing value with its feature mean.

Data cleaning involves also handling noisy data, where noise is identified to be a random error or variance in a measured variable [17].

2.7.2 Data integration

Integration combines data from multiple sources to form a coherent data store. Metadata, correlation analysis, data conflict detection, and the resolution of semantic heterogeneity contribute toward smooth data integration

2.7.3 Data Reduction

Even after the data have been cleaned up, there may still be noise that is irrelevant to the problem being analyzed. These noise attributes may confuse subsequent data mining steps, produce irrelevant rulesand associations, and increase computational cost. It is therefore wise to perform a dimension reduction or feature selection to separate irrelevant attributes.

2.7.4 Data Transformation

In data transformation, the data are transformed into forms appropriate for data mining. Data transformation routines convert the data into appropriate forms for mining. For example, attribute data may be normalized so as to fall between a small rang, such as 0.0 to 1.0. One of the types of data transformation is data normalization in which data are scaled so as to fall in a small specified range. Here are one of the simple and effective normalization techniques:

2.8 DATA MINING TASKS

Data mining tasks usually falls in one of these two categories [20]. Description and prediction. Prediction uses supervised learning technique to predict values of data using known values found from different data. Data mining tasks for prediction include classification, regression, time-series analysis. Description focuses on employing unsupervised learning technique to find human-interpretable patterns describing the data. Data mining tasks for description are clustering, summarization, mining associations, and sequence discovery. The typical and basic data mining tasks involve [21].

1. Association rule mining.
2. Classification.
3. Clustering

2.8.1 Classification

Given a set of objects, each of which belongs to a known class, and each of which has a known vector of variables, classification techniques aim to construct a rule which will allow to assign future objects to a class, given only the vectors of variables describing the future objects [39].

as shown in figure (2.4). Problems of this kind, called problems of supervised Classification deals with categorical class labels. If the class label is numeric, the problem is called regression or numeric prediction [41].

Examples of classification tasks are

- Determining whether a particular credit card transaction is fraudulent.
- Diagnosing whether a particular disease is present.

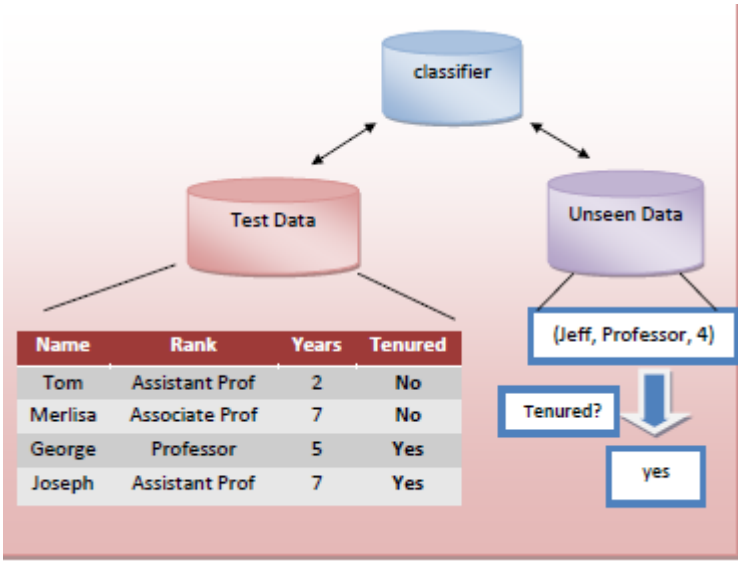


Figure 2.4: Classification process

2.8.2 Clustering

This is also called unsupervised learning. Here, we are given a database of objects that are usually without any predefined categories or classes. We are required to partition the objects into subsets or groups such that elements of a group share a common set of properties. Moreover the partition should be such that the similarity between members of

the same group is high and the similarity between members of different groups is lo [5]. Similarity can be determined using various distance functions [6]. Some of these measures are [16].

2.9 SAMPLE DATA MINING ALGORITHMS

In the IEEE International Conference on Data Mining (ICDM) in December 2006, top 10 algorithms for data mining were identified: C4.5, *k*-Means, SVM, Apriori, EM, PageRank, AdaBoost, *k*NN, Naive Bayes, and CART. These top 10 algorithms are among the most influential data mining algorithms in the research community [38].

We chose to explain three of these algorithms in order to be used later in the proposed data mining systems.

2.9.1 Apriori Algorithm for Mining Association Rules

Apriori is a seminal algorithm proposed by R. Agrawal and R. Srikant in 1994 for mining frequent itemsets for Boolean association rules. The name of the algorithm is based on the fact that the algorithm uses prior knowledge of frequent itemset properties, as we shall see later. By convention, Apriori assumes that items within a transaction or itemset are sorted in lexicographic order. The number of items in an itemset is called its *size* and an itemset of size *k* is called a *k*-itemset. Let the set of frequent itemsets of size *k* be and their candidates be Both and maintain a field, support count. Apriori employs an iterative approach known as a level-wise search, where k-itemsets are used to explore (k+1)-itemsets. To improve the efficiency of the level-wise generation of frequent itemsets, an important property called the Apriori property, presented below, is used to reduce the search space.

- Apriori property all nonempty subsets of a frequent itemset must also be frequent.

This property belongs to a special category of properties called antimonotone in the sense that if a set cannot pass a test, all of its supersets will fail the same test as well. It is called antimonotone because the property is monotonic in the context of failing a test.

2.9.2 Naïve Bayes Algorithm for Data Classification

Bayesian classifiers are statistical classifiers. They can predict class membership probabilities, such as the probability that a given tuple belongs to a particular class. Bayesian classification is based on Bayes' theorem Studies considered that *naïve* Bayesian classifier is comparable in performance with decision tree and selected neural network classifiers. Bayesian classifiers have also exhibited high accuracy and speed when applied to large databases. Naïve Bayes method is important for several reasons. It is very easy to construct, not needing any complicated iterative parameter estimation schemes. This means it may be readily applied to huge data sets. It is easy to interpret, so users unskilled in classifier technology can understand why it is making the classification it makes. And finally, it often does surprisingly well. Naïve Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence. It is made to simplify the computations involved and, in this sense, is considered "naïve".

- Bayes' Theorem Bayes' theorem is named after Thomas Bayes, a nonconformist English clergyman who did early work in probability and decision theory during the 18th century. The class label attribute, buys computer, has two distinct values (namely, {yes, no}).

Table 2.1: Electronic store Database

Age	Income	Student	credit-rating	buy-computer
Youth	High	No	Fair	No
Youth	High	No	Excellent	No
Middle	High	No	Fair	Yes
Senior	Medium	No	Fair	Yes
Senior	Low	Yes	Fair	Yes
Senior	Low	Yes	Excellent	No
Middle	Low	Yes	Excellent	Yes
Youth	Medium	No	Fair	No
Youth	Low	Yes	Fair	Yes
Senior	Medium	Yes	Fair	Yes
Youth	Medium	Yes	Excellent	Yes
Middle	Medium	No	Excellent	Yes

X = (age = youth, income = medium, student = yes, credit rating = fair) We need to maximize (|) ( ), for i = 1, 2. ( ), the prior probability of each class, can be computed based on the training tuples:

P(buys computer = yes) = 9/14 = 0.643

P(buys computer = no) = 5/14 = 0.357

To compute (|), for i = 1, 2, we compute the following conditional probabilities:

P(age = youth / buys computer = yes) = 2/9 = 0.222 P(age = youth / buys computer = no) = 3/5 = 0.600

P(income = medium / buys computer = yes) = 4/9 = 0.444

P(income = medium / buys computer = no) = 2/5 = 0.400

P(student = yes / buys computer = yes) = 6/9 = 0.667

P(student = yes / buys computer = no) = 1/5 = 0.200

P(credit rating = fair /buys computer = yes) = 6/9 = 0.667

P(credit rating = fair / buys computer = no) = 2/5 = 0.400

Using the above probabilities, we obtain

P(X| buys computer = yes) = P(age = youth | buys computer = yes) * P(income = medium |buys computer = yes) *

P(student = yes | buys computer = yes) * P(credit rating = fair |buys computer = yes)

= 0.222*0.444*0.667*0.667 = 0.044. Similarly,

P(X| buys computer = no) = 0.600*0.400*0.200*0.400 = 0.019.

To find the class, that maximizes we compute

P(X| buys computer = yes)P(buys computer = yes) = 0.044*0.643 = 0.028 P(X| buys computer = no)P(buys computer = no) = 0.019*0.357 = 0.007 Therefore, the naïve Bayesian classifier predicts buys computer = yes for tuple X.

2.9.3 K-means Algorithm for Data Clustering

The k-means algorithm is a simple iterative method to partition a given dataset into a user specified number of clusters, k. This algorithm has been discovered by several researchers across different disciplines, most notably Lloyd (1957, 1982), Forgey (1965), Friedman and Rubin (1967), and McQueen (1967).or “centroids”. Then the algorithm iterates between two steps till convergence:

- Step 1: Data Assignment. Each data point is assigned to its closest centroid, with ties broken arbitrarily. This results in a partitioning of the data.
- Step 2: Relocation of "means". Each cluster representative is relocated to the center (mean) of all data points assigned to it. If the data points come with a probability measure (weights), then the relocation is to the expectations (weighted mean) of the data partitions. The algorithm can use any of the distance measures. The algorithm converges when the assignments (and hence the values) no longer change. The algorithm execution is visually depicted in the example. Note that each iteration needs $N \times k$ comparisons, which determines the time complexity of one iteration. The number of iterations required for convergence varies and may depend on N, but as a first cut, this algorithm can be considered linear in the dataset size depicts the algorithm steps:

Algorithm: *k*-means.

Input:

k: the number of clusters,

D: a data set containing *n* objects.

Output: A set of *k* clusters.

Method:

(1)arbitrarily choose *k* objects from D as the initial cluster centers;
(2)repeat
(3)(re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
(4)update the cluster means, i.e., calculate the mean value of the objects for each cluster;
(5)until no change;

Table 2.2: Example Database

id	**X**	**Y**
1	2	10
2	2	5
3	8	4
4	5	8
5	7	5
6	6	4
7	1	2
8	4	9

2.10 VISUAL DATA MINING

Visual data mining a highly attractive and effective tool for the comprehension of data distributions, patterns, clusters, and outliers in data. Data visualization: Data in a database or data warehouse can be viewed at different levels of granularity or abstraction, or as different combinations of attributes or dimensions. Data can be presented in various visual forms, such as boxplots, 3-D cubes, data distribution charts, curves, surfaces, link graphs, and so on Visual display can help give users a clear impression and overview of the data characteristics in a database.

2.10.1 Data Mining Result Visualization

Visualization of data mining results is the presentation of the results or knowledge obtained from data mining in visual forms. Such forms may include scatter plots and boxplots (obtained from descriptive data mining), as well as decision trees, association rules, clusters, outliers, generalized rules, and so on.

2.10.2 Data Mining Process Visualization

This type of visualization presents the various processes of data mining in visual forms so that users can see how the data are extracted and from which database or data warehouse they are extracted, as well as how the selected data are cleaned, integrated, preprocessed, and mined. Moreover, it may also show which method is selected for data mining, where the results are stored, and how they may be viewed.

2.11 EXAMPLES OF COMMERCIAL DATA MINING SYSTEMS

Although data mining is a young and new field, many data mining systems products are available in markets. Here is a summary of some publically available data mining products [16].

- Clementine, from SPSS, provides an integrated data mining development environment for end users and developers. Multiple data mining functions, including association mining, classification, prediction, clustering, and visualization tools, are incorporated into the system. A distinguishing feature of Clementine is its object oriented, extended module interface, which allows users' algorithms and utilities to be added to Clementine's visual programming environment.
- Enterprise Miner was developed by SAS Institute, Inc. It provides multiple data mining functions, including association mining, classification, regression, clustering, time series analysis, and statistical analysis packages. A distinctive feature of Enterprise Miner is its variety of statistical analysis tools, which are built based on the long history
- CART, available from Salford Systems, is the commercial version of the CART (Classification and Regression Trees) system. It creates decision trees for classification

and regression trees for prediction. CART employs boosting to improve accuracy. Several attribute selection measures are available.

- Weka, developed at the University of Waikato in New Zealand, is open-source data mining software in Java. It contains a collection of algorithms for data mining tasks, including data preprocessing, association mining, classification, regression, clustering, and visualization. Many other commercial data mining systems and research prototypes are also fast evolving.
- IBM Intelligent Miner: Intelligent Miner is an integrated and comprehensive set of data-mining tools. It uses decision trees, neural networks, and clustering. The latest version includes a wide range of text-mining tools. Most of its algorithms have been parallelized for scalability. A user can build models using either a GUI or an API.
- DBMiner DBMiner is a publicly available tool for data mining. It is a multiple-strategy tool and it supports methodologies such as clustering, association rules, summarization, and visualization. DBMiner uses Microsoft SQL Server 7.0 Plato and runs on different Windows platforms.

2.12 DATA MINING APPLICATIONS

As a young research field, data mining has made broad and significant progress since its early beginnings in the 1980s.Today, data mining offers value across a broad spectrum of industries telecommunications and credit card companies are two of the leaders in applying data mining to detect fraudulent use of their services. Insurance companies and stock exchanges are also interested in applying this technology to reduce fraud. Medical applications are another fruitful area: data mining can be used to predict the effectiveness of surgical procedures, medical tests or medications. Companies active in the financial marketing use data mining to determine market and industry characteristics as well as to predict individual company and stock performance. Retailers are making more use of data mining to decide which products to stock in particular stores(and even how to place them within a store),as well as to assess the effectiveness of promotions and coupons. Pharmaceutical firms are mining large databases of chemical compounds and of genetic material to discover substances that might be candidates for development as agents.

3. DATA MINING AND AGENT

Through information processing (1970–80) to information environments (1990–2000). Information environments, such as the worldwide web provide the basis for *autonomous* software systems. Examples of such systems are systems for mail and news delivery, document indexing (web crawlers), and consistency maintenance. Autonomous systems have also appeared in robotics, in particular for mobile robots. At the same time, software is becoming complex to the point of being unmanageable. For example, the Microsoft Windows 2000 system contains 35 million lines of code with many bugs. Its release date has been pushed back several times and it is not clear whether it will ever become sufficiently reliable for commercial success. To avoid the difficulties of maintaining consistency in such large software projects, the tendency is to decompose systems into small components which can be understood independently of each other. However, component paradigms are still lacking intuitive metaphors which make component behavior easy to understand. Both developments – autonomous software and robotic agents as well as software components – are now converging into a single technology: *agents*. Agents turn software components into proactive processes [8].

Agent-based computing has been hailed as ‘the next significant breakthrough in software development’, and ‘the new revolution in software. Currently, agents are the focus of intense interest on the part of many sub-fields of computer science and artificial intelligence. Agents are being used in an increasingly wide variety of applications, ranging from comparatively small systems such as email filters to large, open, complex, mission critical systems such as air traffic control. At first sight, it may appear that such extremely different types of system can have little in common. And yet this is not the case: in both, the key abstraction used is that of an agent [15].

3.1 AGENTS AND AGENT-BASED SYSTEMS

In particular, there is no real agreement even on the core question of exactly what an agent is [15].

There is much confusion about what people mean by agent.the meaning of term agent, can change emphasis when an alternative perspective is applied and this can lead to

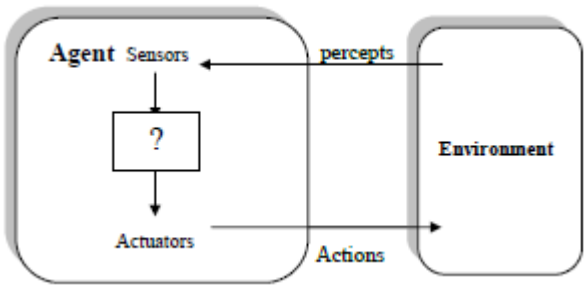
confusion. People will also often tend to use the definition they are familiar with from their own background and understanding [34].

Woodridge and Jennings defined agent as a computer system situated in some environment, and that is capable of *autonomous action* in this environment in order to meet its design objectives [15].

From AI perspective, an agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators [28].

By an agent-based system, we mean one in which the key abstraction used is that of an agent. Note that an agent-based system may contain any non-zero number of agents. The multi-agent case where a system is designed and implemented as several interacting agents, is both more general and significantly more complex than the single-agent case. However, there are a number of situations where the single-agent case is appropriate. A good example, as we shall see later in this chapter, is the class of systems known as expert assistants, wherein an agent acts as an expert assistant to a user attempting to use a computer to carry out some tasks [15].

3.2 AGENTS AND ENVIRONMENTS



,

Figure 3.1: Agent and Environment

3.2.1 Perception of Agent

In order for a software agent to take some action, it first has to be able to perceive what is going on around it, to have some idea of the state of the world. For animals, this problem is solved by the senses of touch, smell, taste, hearing, and sight. The next problem is not to get overwhelmed by the constant stream of information. Because of the large amount

of raw sensory input we get, one of the first things humans learn is to filter out and ignore inputs that are expected or usual. We develop an internal model of the world, of what the expected consequences are when we take an action. As long as things are going as expected, we can get by without paying too much attention (for example, someone driving a car while talking on a cellular phone). But we also have learned to focus immediately on unexpected changes in the environment, such as a car appearing in our peripheral vision while entering an intersection. If our agent works in the e-mail or newsgroup monitor domain, it will have to recognize when new documents arrive, whether the user is interested in the subject matter or not, and whether to interrupt the user at some other task to inform him of the newly available information. All of this falls into the realm of perception. Of course, we can design our agents and their messages in such a way that they don't have to be *very* perceptive. We can require the user to explicitly tell our agent what to do and how to do it. The events that are generated could contain all the information the agent needs to determine the current state and the appropriate action [3].

3.2.2 Action of Agent

Once the agent has perceptively recognized that a significant event has occurred, the next step is to take some action. This action could be to realize that there is no action to take, or it could be to send a message to another agent to take an action on our behalf. Like people, agents take actions through effectors. For people, an effector is our muscles, when we take physical action in the world. Or we can take action through speech (using different muscles) or by sending e-mail (different muscles again). By communication with other people, we can cause them to act and change the environment.

Table 3.1: Shows some examples of agents and their corresponding environments, sensors

Agent Type	Environment	Actuators	Sensors
Medical diagnosis system	Patient, hospital, Staff	Display questions, tests, diagnoses, treatments, referrals	Keyboard entry of symptoms, findings, patient's answers
Satellite image analysis system	Downlink from orbiting satellite	Display categorization of scene	Color pixel arrays
Part-Picking Robot	Conveyor belt with parts; bins	Jointed arm and hand	Camera, joint angle sensors
Refinery Controller	Refinery, operators	Valves, pumps, heaters, displays	Temperature, pressure, chemical sensors

3.3 AGENT PROPERTIES

Franklin and Grasser listed several properties that may help us further classify agents in useful ways [10].

These properties are shown in table (3.2).

Franklin and Grasser consider the first four items in the table to be the minimum requirements for an agent, i.e. an agent should be able to react autonomously and goal-directed to signals in their environment.

Table 3.2: Properties of Agents

Reactive(sensing)	Responds in a timely fashion to changes in the Environment
Autonomous	Exercises control over its own actions
Goal-oriented(purposeful)	Does not simply act in response to the environment
Temporally Continuous	Is a continuously running process
Communicative	Communicates with other agents, perhaps including people
Learning(adaptive)	Changes behavior based on previous experience

There are, of course, other possible classifying schemes. For example, we might classify software agents according to the tasks they perform, for example, information gathering agents or email filtering agents. Or, we might classify them according to their control architecture. Agents may also be classified by the range and sensitivity of their senses, or by the range and effectiveness of their actions, or by how much internal state they possess [10].

3.4 AGENT'S BEHAVIOR

The way an agent behaves is often used to tell them apart and to distinguish what and who they are, whether animal, human or artificial. Behaviors range from the fully conscious (cognitive) to the unconscious (reactive) [35].

3.4.1 Reactive Agents

Reactive agents are very limited in what they can do as they do not have the ability to plan, coordinate between themselves or set and understand specific goals; they simply react to events when they occur. This does not preclude them from having a role to play in producing intelligent behavior. The reactive school of thought is that it is not necessary for agents to be individually intelligent. However, they can work together collectively to solve complex problems.

3.4.2 Cognitive Agents

The cognitive school of thought seeks to build agents that exhibit intelligence in some manner. In this approach, individual agents have goals, and can develop plans on how to achieve them. They use more sophisticated communication mechanisms, and can intentionally coordinate their activities. They also map their environment in some manner using an internal representation or knowledge base that they can refer to and update through learning mechanisms in order to help guide their decisions and actions. As a result, they are much more flexible in their behavior compared to reactive agents.

Table 3.3: Reactive Agents versus Cognitive Agents

Reactive Agents	**Cognitive Agents**
Use simple behaviors	Use complex behaviors
Have low complexity	Have high complexity
Are not capable of foreseeing the	Anticipate what is going to happen
Future	
Do not have goals	Have specific goals
Do not plan or coordinate amongst	Make plans and coordinate with each
Themselves	Other
Have no representation of the	Map their environment
Environment	
Do not adapt or learn	Exhibit learned behavior
Can work together to resolve complex	Can resolve complex problems both by
Problems	working together and by working
	Individually

3.5 DATA MINING PROCESS AND AGENTS

In several steps through knowledge discovery, which include data preparation, mining model selection and application, and output analysis, agent paradigm can be used to automate the individual tasks. In data preparation, agent use can be especially on sensitivity to learning parameters, applying some triggers for database updates and handling missing or invalid data. In data mining model, we have seen the agent-based studies are implemented for classification, clustering, summarization and generalization which have learning nature and rule generation since current learning methods are able to find regularities in large data sets. An intelligent agent can use domain knowledge with embedded simple rules and using the training data it can learn and reduce the need for domain experts. In the interpretation of what is learned, a scanning agent can go through the rules and facts generated and identify items that can possibly contain valuable information. Data preparation in data mining involves data selection, data cleansing, data preprocessing, and data representation. With the use of intelligent agents, several of these steps can possibly be automated.

3.6 USING JAVA FOR DATA MINING AND AGENTS

Using Java language for data mining and agent-based systems can be seen from two perspectives:

3.6.1 Java for Data Mining

The Java language is considered a good tool for building database applications, since Java Database Connectivity (JDBC) provides an easy and powerful way for building complete database applications. JDBC offers a standard way for accessing data and dealing with different database management systems. So, by using JDBC , the developer can add, change, delete or retrieve data using SQL instructions from inside Java applications. JDBC is the interface that the developer can see the database from. In addition, JDBC extends Java independency notion, since it allows changing the database and even changing the database management system without any need to change the programming code. All what we need is to change the JDBC driver.

4. DESIGN OF THE AGENT

As it was mentioned in chapter 1, many researchers agreed that data mining is a difficult and laborious activity that requires a great deal of expertise and a highly trained professionals to do an iterative, multistep process Each stage in data mining is very huge and consists of a lot of methods and algorithms. For example preprocessing includes cleaning, integration, transformation, and many others. Following it in the next step, data mining methods have the same problem. Data mining includes many tasks like classification clustering association, and others. Each one of these tasks includes a lot of algorithms, each with its own way of implementation. A suggested solution to this fact is using agents to automate the different steps and take important decisions instead of users. So, in the proposed work, we suggested to design and implement a prototype Data Mining Agent System or (DMAS) for abbreviation Borland Jbuilder9, a Java language development environment, was used for implementing the system. Section 4.2 explains the design of the proposed DMAS, while section 4.3 shows the implementation.

4.1 DESIGN OF THE PROPOSED DMAS

The structure of the proposed system gathered between the agent architecture and the data mining process stags. This integration resulted in the system .The overall system process can be viewed as receiving a database as an input, and passing it in the different mining stages according to the database analysis results, where the analysis results depend mainly on the database metadata accordingly ,the software agent can choose the suitable mining task. In the work we used three most popular mining tasks these are classification, clustering and association mining. The designed agent is considered as a reactive agent, or a simple reflex agent since it receives a sensory input, apply the agent function which includes performing the data mining stages and taking decisions concerning the preprocessing needs and the suitable mining task depending on a set of condition-action rules, and finally generating the output.

4.1.1 Agent's Sensory Input

Software Agents usually receives keystrokes, file contents or network packets as sensory input. The agent in DMAS receives the database file and its metadata file as an input. The database file is a set of transactions or a table where rows are records, and columns are

the attributes .While the metadata file contains descriptive information about the attributes, their names, types and values.B Since "metadata is used by analysts in understanding data and building models and it can support the selection of suitable mining methods Then it is important for the agent to depend on it in the stages of its work.

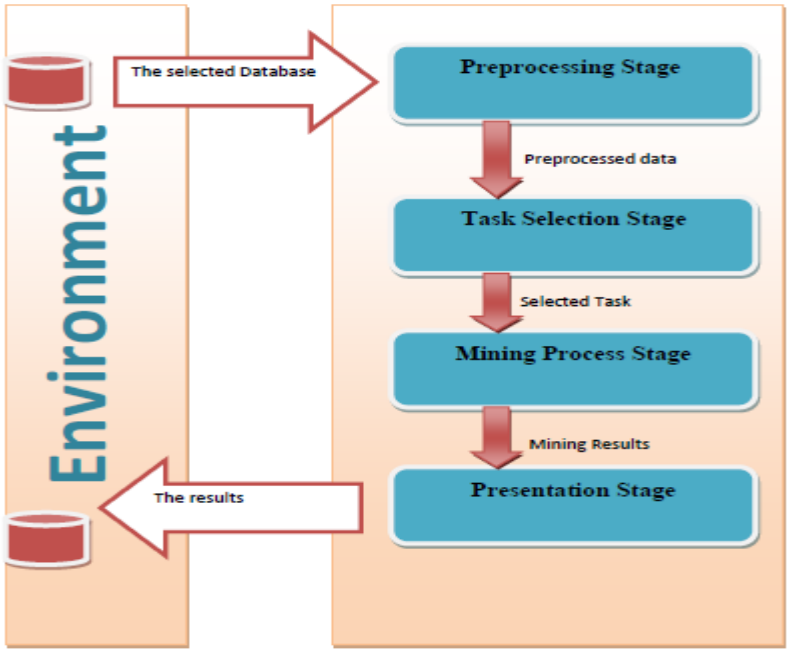


Figure 4.1: Block Diagram of The Proposed System

4.1.2 Agent's Function

As it was shown in Figure (3.1), agents deal with environments through by sensing an input, applying a function and giving an action. The agent function (Figure 4.2) in the proposed system is to handle the work of the data mining system automatically and execute its different stages depending on a set of condition-action rules.

Algorithm : Agent Function

Input : Database Files + metadata files

Output : Knowledge base file

Each stage in the above algorithm will be described in details in the following subsections. In this section of the work, the system applied the preprocessing techniques required. There are three important techniques performed:

A. Descriptive Data Summarization:

One of the mathematical measures found in the descriptive data summarization techniques for measuring the central tendency of some data is the mean:

B. Transformation:

The proposed system used normalization to let numeric attribute values fall in a small range.The method used for normalization is min-max normalization

C. Cleaning:

Cleaning includes many strategies to handle missing values and noisy data. The proposed system has the ability to handle missing values through the following methods:

- If the attribute type is categorical:
- If the attribute type is numeric:

Use the attribute mean to fill in the missing value.

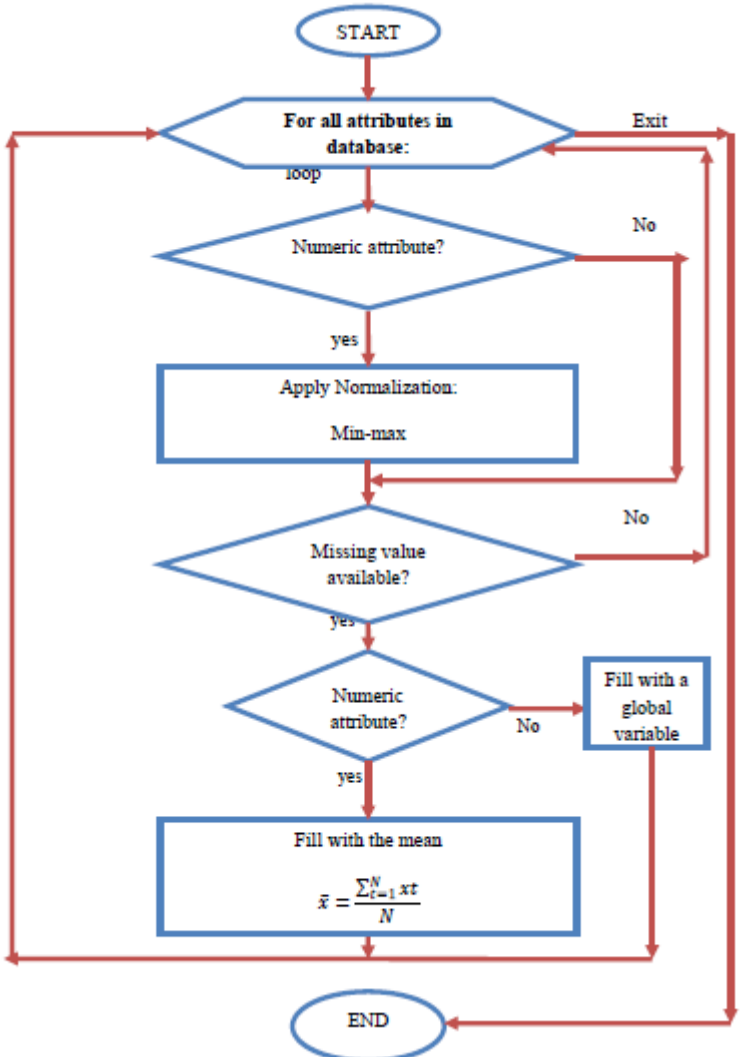


Figure 4.2: Flowchart of the preprocessing stage

4.2 TASK SELECTION STEP

In this section, the database is ready to undergo the mining step, after it had been preprocessed.The proposed system includes a suite of three data mining tasks among the most popular and important ones:

- Classification
- Clustering
- Association Mining

For the current fetched database the system chooses the best task automatically by the software agent which depends on a set of predefined rules. As it was mentioned in

chapter 2, data mining tasks falls into two types descriptive and predictive. Classification is a descriptive method, while clustering and association mining are predictive ones. Since the main difference between both types is the presence of a target attribute or class in the database, then it is easy to characterize databases that need to be classified. In addition, whereas Association mining takes transactional databases with one attribute, Clustering accepts many attributes with different types. Under this set of condition-action rules, the agent will analyze the meta data information

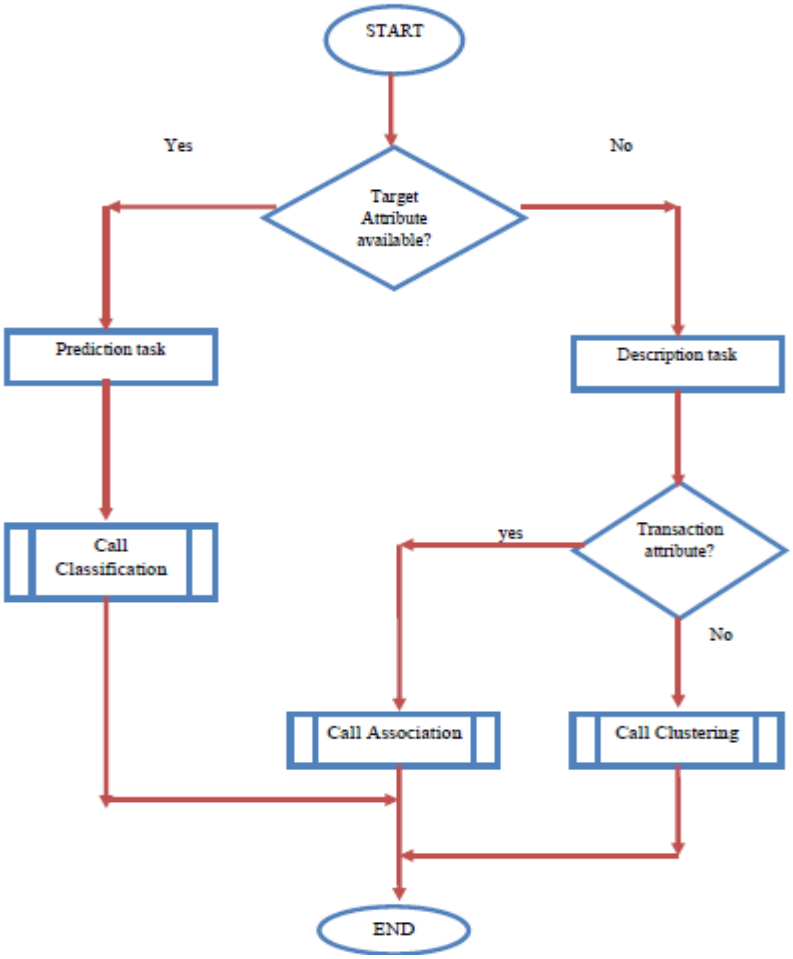


Figure 4.3: Flowchart of Task Selection

4.3 MINING STEP

After the suitable DM task has been chosen, the work now is to apply the corresponding algorithms. The proposed system calls bayseian algorithm for classification, k-mean algorithm for clustering, and apriori algorithm for association mining.

A. Apriori Algorithm:

Apriori algorithm is one of the algorithms including in mining frequent itemsets for association rules and has been implemented in the system. Apriori algorithm is illustrated in section 2.9.1.The input and ın the implementation of Apriori, the values of minconf and minsup were prespecified.Apriori algorithm generates frequent itemsets stored in arrays, then we implement genrules function to generate rules from frequent itemsets. The resulted frequent itemsets enter another procedure to generate association rules depending on minimum confidence. Input: frequent itemsets the resulted association rules are stored in the knowledgebase file.

B.Naïve Bayes Algorithm

Naïve Bayes algorithm is one of the algorithms in the classification task and it is illustrated in section 2.9.2. The input and output of the algorithm: Input: Training database and test tuple In the proposed system, in the case where the agent select the classification task for a given database, it considers this database as the training one and ask for test datasets to give answers. The test data undergo set of calculations to estimate probabilities according to bayesian algorithm. The resulted class is stored absolutely in the knowledgebase file.

C. K-mean Algorithm

k-mean algorithm is one of the algorithms of the clustering task. The input and output of k-mean algorithm are as the following: Input: Database file, k.. In the implementation of the proposed system, the value of k was suggested to be equal to three and the distance measure used is the city block distance The obtained k clusters are stored in the knowledge base file.

4.4 RESULT PRESENTATION STEP

This section of the system deals with the software agent that can be presented by three kinds. This step should reflect the real work and effort was made by the agent and give us an idea about the series of steps that were automated. In the proposed system, we always obtain one of these three kinds of results: In order to present results in a complete form, the agent behaves in two ways:Storing results in a knowledge-base file: The agent stores the obtained results (clusters, rules or class value) in a file with the following information: Database name, Table name, Number of attributes and records, and Obtained results Presenting results on the interface: The proposed system present the selected task and the agent steps on the user interface to let the user follow and understand the work. Finally, the agent will be searching for another database files in the environment.

4.4.1 Agent's Output

The agent's output is the knowledge base file containing the different kinds of results.

4.4.2 Implementation of DMAS in Java

Java language environment was used for implementing the proposed DMAS. Reasons behind using it are illustrated in section (3.7). In order to connect with a database, Java provides many JDBC drivers according to the type of databases. We used the JDBC-ODBC driver due to its simplicity, ease of use, and availability with the different java environments, so it does not need any previous preparation. The main classes used are:The attribute class data are:

- Attribute.name denotes the attribute name, for example: Race.
- Attribute.type denotes the type of the attribute whether it is discrete (categorical, nominal) or continuous (numeric).
- Attribute. is an array of the allowed variables for the discrete attribute types only. For example the (Race) attribute accept five values which are: [White, Black, Eskimo, Islander, others]
- Attribute.v stores the number of the discrete values stored in attribute.value

- Datasource: stores the name of the current database
- Tablename: stores the of table in the current database.
- Filename: stores the name of file storing metadata information.
- array of instances of the class attribute storing the attributes names, types, and values

A) A counter for the array denotes number of attributes. DMAgent the default constructer, it calls the opendbstore method. Opendbstore Open the file that stores the database name, table names, and metadata file names, which is considered as an environment, and search for the presence of any databases. Preparemeta Opens the metadata file, reads its contents, and stores them in instances of preprocess this method applies some preprocessing techniques if needed. Present Opens the database file, reads its contents and show them in the JTable component added in the designed user interface. Taskselection Executes the steps illustrated in the flowchart run this method represents the agent function. It calls the other methods found in the class and determine the sequence of stages automated by the agent.This method generates the candidate itemsets by performing the join and prune step. The following code depicts the main steps in the method.

```
Public table apriorigen(table[] L, int k){…

// join step

for(j1=0;j1 < bound-1 ;j1++)

for(j2=j1+1;j2 < bound;j2++)

{

boolean f1=true;

for(i=0;i<=k-2;i++)

if (L[k-1].m[j1].set[i] != L[k-1].m[j2].set[i])

f1=false;
```

```
if ((f1==true)&&(L[k-1].m[j1].set[k-1]<L[k-1].m[j2].set[k-1])){ for(i=0;i<=k-1;i++)

join[i]=L[k-1].m[j1].set[i];

join[k]=L[k-1].m[j2].set[k-1];

// prune step

if (check(join,k)==true)

{ for(int a=0;a<=k;a++) {c[k].m[jc].set[a]=join[a]; System.out.print(c[k].m[jc].set[a]);}
System.out.println();

jc++;}

}}

}
```

3. Check(join,k):

This method is called in the prune step to check if all the itemsets generated in the join step verify the apriori property.

If all the subsets of the itemset are frequent then it can be added to the frequent itemsets , and the method return true, otherwise the itemset should be pruned and the method returns false, as illustrated

- Apriori (minsup):The first step for this method is to find the frequent 1-itemset by calling find1()method. Then it calls apriorigen function to generate the K+1 candidates. After that it scans the file to calculate their support counts, and determine the frequent itemsets depending on the minimum support.
- Genrules(k): Scans all the generated frequent itemsets and calls the method rules() for each one.
- Retsup(sub,s):

Takes any itemset and returns its support count stored in the tables.

- Run():

The run method calls apriori() method to generate all frequent itemsets with their counts, and then calls genrules() method to

generate rules from them:

```
public void rrun() {

int minsup=15;

int k=Apriori(minsup);

genrules(k);

}//
```

- Naïve Bayes Class: Baysian class data array of instances of the class attribute storing the attributes names, types, and values

aa: a counter for the array(att[]) denotes number of attributes. Baysian class methods are:

Baysian(): the default constructer

This method performs the main steps of the baysian algorithm.

```
public void work()

// Enter test Data

System.out.println("before input");

for(int i=0;i<aa-1;i++){

x[i]=JOptionPane.showInputDialog(null,"enter "+att[i].name);

System.out.println("after input");
```

```
// For all records in the database

while(rs.next() ){

total++;

s=rs.getString(att[z].name);

// For all attributes: calculate the posterior probability P(xk|ci)

for(int i=0;i<aa-1;i++)

g=rs.getString(att[i].name);

If (g.equals(x[i])){

for(int j=0;j<att[z].v;j++)

if (s.equals(att[z].val[j]))

pxc[i][j]++;

}

//Calculate the prior probability P(Ci)

for(int j=0;j<att[z].v;j++)

if(s.equals(att[z].val[j]))

c[j]++;

}

}

for(int j=0;j<att[z].v;j++)
```

```
{ pc[j]=c[j]/total; System.out.println("c/t=p"+c[j]+"/"+total+"="+pc[j]);

for(int i=0;i<aa-1;i++)

for(int j=0;j<att[z].v;j++)

pxc[i][j]=pxc[i][j]/c[j];

// Calculate posterior probability P(X|Ci)*P(Ci)

for(int j=0;j<att[z].v;j++)

{ u[j]=1;

for(int i=0;i<aa-1;i++)

u[j]=u[j]*pxc[i][j];}

for(int j=0;j<att[z].v;j++)

u[j]=u[j]/pc[j];

}//end main
```

- K-means Class:: att[]: array of instances of the class attribute storing the attributes names, types, and values .

aa: a counter for the array(att[]) denotes number of attributes. Total: stores the total number of objects in the database.K-means class methods are: Kmeans(): the default constructer.Opensource Opens the database file, and determine only the continuous attributes since k-mean algorithm deals only with the continuous attributes.

5. EXPERIMENTAL RESULTS

To accomplish the test of the designed system, Different types of databases with their metadata information were downloaded from different websites, some of which are professional in data mining research datasets

5.1 DATASETS

5.1.1 Database 1

The first database is the cereal database is available online with its

Attribute description at the following hyperlink http://lib.stat.cmu.edu/DASL/Stories/HealthyBreakfast.html . Part of the dataset is presented in Appendix A . This database illustrates different kinds of cereals and their nutritional composition. The metadata information about the attributes and their types is shown in table (5.1). The agent fetched this database from the environment and then started applying the four designed stages. The process steps done by the agent can be presented in the following interface Figure (5.1). The first step for the agent after loading the database was to check any need for normalization or handling missing values and apply them if necessary. Secondly, the agent should decide which mining task or method to apply. For this reason the agent scanned the metadata file for attribute types, and it decided to choose the clustering task depending on the rules illustrated in flowchart 4.3). The designed system stored the resulted clusters from the k-mean method in a knowledgebase file.

5.1.2 Database 2

The second database is adult.mdb. It is available online on the Machine Learning Repository UCI at the following hyperlink: http://archive.ics.uci.edu/ml/datasets/Adult.

Table 5.1: Metadata of Database

Attribute Name	**Attribute Type**	**Attribute values**	
Workclass	Discrete	Private, Self-emp-not -inc, Self-	
		emp-inc, Federal-gov, Local-gov,	
		State-gov, Without-pay, Never-	
		worked.	
Education	Discrete	Bachelors, Some-college, 11th, HS-	
		grad, Prof-school, Assoc-acdm,	
		Assoc-voc, 9th, 7th-8th, 12th,	
		Masters, 1st-4th, 10th, Doctorate,	
		5th-6th, Preschool.	
Marital_status	Discrete	Married-civ-spouse,	Divorced,
		Never-married,	Separated,
		Widowed, Married-spouse-absent,	
		Married-AF-spouse.	
Occupation	Discrete	Tech-support, Craft-repair, Other-	
		service, Sales, Exec-managerial,	
		Prof-specialty, Handlers-cleaners,	
		Machine-op-inspct, Adm-clerical,	
		Farming-fishing, Transport-moving,	
		Priv-house-serv,	Protective-serv,
		Armed-Forces.	
Relationship	Discrete	Wife, Own-child, Husband, Not-in-	
		family, Other-relative, Unmarried.	

The agent checked for any preprocessing needs. Then it scanned the metadata file for attribute types, it decided to choose the classification task due to the presence of a target attribute and according to the conditions presented in flowchart(4.3).

Since the task is classification, the agent considered the loaded database as a training set and asked the user to enter his test data, in order to be classified. The following figure (5.3) show the agent messages sent to the user to input his data.

5.1.3 Database 3

The third database is mushroom.txt. It is available online on the following hyperlink http://fimi.ua.ac.be/data/. Part of the dataset is presented in the Appendix sainces the database contain only transactions of items, the metadata information denotes that the type of the database which is transactional. After the agent scanned the metadata file, it found that it is transactional dataset and calls Apriori algorithm.

5.1.4 Database 4

The fourth database is car.mdb and is available online on the Machine Learning Repository UCI at the following hyperlink: http://archive.ics.uci.edu/ml/datasets/carevaluation. Part of the dataset is presented in the

Table 5.2: Metadata of database

Attribute Name	Attribute Type	Attribute values
Buying(buying price)	Discrete	vhigh, high, med, low
Maint(maintainace price)	Discrete	vhigh, high, med, low
Doors(number of doors)	Discrete	1,2,3,4,5more
Persons(capacity to carry	Discrete	2,4,more
persons)		
Lugboot(size of luggage boot)	Discrete	small, med, big

5.2 CONCLUSIONS

Designing and implementing the DMAS resulted in several conclusions:

- Concerning the benefits of the system:
  The designed system facilitates the work of both experts and non-expert users

Non-experts can do things that cannot do usually. Experts can do things more easily and without effort. The proposed system does not need user interference or help since it is designed to work on behalf of user completely. Data mining and agents can integrate together in many ways and forms to give many benefits concerning automation, simplicity and performance and this motivates us for using more intelligent and complicated agent architectures to verify more advanced goals.

- Concerning the implementation part:

For apriori algorithm: Storing itemsets in classes and arrays as it was implemented is very effective and easy to understand way, but it is time and memory consuming especially for large datasets, so it can be replaced this way by storing them in files.

Although baysian algorithm is strong and easy as it was mentioned previously, but just classifying the test tuple without generating general classification rules or decision tree makes it less interesting and minimizes its benefits.

- Concerning the design part: By comparing the proposed DMAS with the typical data mining system architecture depicted in section(2.4) we find:

- Kind of database: The proposed DMAS deals with a single table or a flat file.
- Database or data warehouse server: In DMAS, the agent is responsible on fetching the data after sensing an input database name.
- Knowledge base**:** The knowledge base in DMAS includes the metadata information that contains information about the database and its attributes. It guides the mining step toward choosing the suitable task.

REFERENCES

[1] Karl Aberer, Data Mining, Laboratoire de systèmes d'informations répartis, 2007.

[2] Kamal Ali Albashiri, Frans Coenen, and Paul Leng, An investigation into the issues of Multi-Agent Data Mining, Doctoral thesis, University of Liverpool, 2007.

[3] Joseph P. Bigus, Jennifer Bigus, Constructing Intelligent Agents with Java, John Wiley & Sons, 2003.

[4] Samo Bobek, Igor Perko, Intelligent Agent based Business Intelligence, Current Developments in Technology-Assisted Education, pp.1047-1051, FORMATEX, 2006.

[5] Macros M. Campos, Peter J. Stengard, and Boriana L. Milenova, Data-Centric Automated Data Mining, International conference on machine learning and applications, 2005.

[6] Vineet Chaoji, Apirak Hoonlor, Boleslaw K. Szymanski, Recursive Data Mining for Role Identification in Electronic Communications, International Journal of Hybrid Intelligent Systems IJHIS, vol.7, no.2, p.p.89-100, 2010.

[7] John Cleary,Geoffrey Holmes, Sally Jo Cunningham, Ian H. Witten, MetaData for Database Mining, University of Waikato, New Zealand,2004.

[8] Boi Faltings, Intelligent Agents: Software Technology for the new Millennium, INFORMATIK • INFORMATIQUE 1 2000.

[9] Usama Fayyad, Gregory Piatetsky, and Padhraic Smyth, From Data Mining to Knowledge Discovery in Databases,

[10] S. Franklin, A. Graesser, Is it an agent, or just a program?, Proceedings Third International Workshop on Agent Theories, Architectures and Languages, Budapest, Hungary, 1996.

[11] Sanjay Garg and Ramesh Ghandra Jain , Variations of K-mean Algorithm:A Study for High Dimensional Large Data Sets, Information Technology Journal, 2006.

[12] Jiawei Han, Micheline Kamber, Data Mining: Concepts and Techniques, Elsevier, 2006.

[13] Jochen Hipp, Ulrich G¨untzer and Gholamreza Nakhaeizadeh, Algorithms for Association Rule Mining – A General Survey and Comparison, SIGKDD Explorations, 2000.

[14] Jochen Hipp, Ulrich Guntzer, and Gholamreza Nakhaeizadeh, Data Mining of Association Rules and the Process of Knowledge Discovery in Databases, Springer, 2002.

[15] N.R. Jennings, M. Wooldridge, Applications of Intelligent Agents, International Agent Technology Conference IAT, 1998.

[16] Mehmed Kantardzic , Data Mining: Concepts, Models, Methods, and Algorithms, John Wiley & Sons, 2003.

[17] Nittaya Kerdprasop, Kittisak Kerdprasop, Moving Data Mining Tools toward a Business Intelligence System, World Academy of Science, Engineering and Technology 25, 2007.

[18] Hussein Keitan Al-Khafajy, Knowledge Discovery in Database by using Data Mining, Ph.D. thesis, Department of Computer Science and Information System of University of Technology, 2002.

[19] Markus Lang, Implementation of Naive Bayesian Classifiers in Java, Kaiserslautern University of Applied Sciences, 2002.

[20] Daneil T.Larose, Discovering Knowledge in Data*:* An Introduction to Data Mining, John Wiley & Sons, 2005.

[21] Daneil T. Larose, Data Mining Methods and Models, John Wiley & Sons, 2006.

[22] P.A.Mitkas, A.L. Symeonidis, D. Kehagias and I. Athanasiadis, Application of Data Mining and Intelligent Agent Technologies to Concurrent Engineering, 10th international conference on concurrent engineering, pp.11-18, Portugal, 2003.

[23] S.Nagabhushana, Data Warehousing, OLAP and Data Mining, New Age International, India, 2006.

[24] Thomas Nocke, Heidrun Schumann, Meta Data for Visual Data Mining, In the conference on computer graphics and imaging CGIM, 2002.

[25] Serge Parshutin and Arkady Borisov, Agents Based Data Mining and Decision Support System, Lecture notes in computer science, volume 5680/2009, pp. 36-49,Springer-Verlag Berlin Heidelberg , 2009.

[26] Gregory Piatetsky-Shapiro, Data mining and knowledge discovery 1996 to 2005: overcoming the hype and moving from university to business and analytics, Data Mining and Knowledge Discovery, vol.15, no.1, pp.99-105, 2007.

[27] Arti Rana, Ashish Jolly, Priyanka Pawar, Use of Intelligent Agents As Data Mining Tools In Knowledge Management AIMT Ambala City, 2004.

[28] Stuart Russell, Peter Norvig, Artificial Intelligence: A modern Approach, Pearson Education, 2003.

[29] Ayse Yasemin Seydim, Intelligent Agents: A Data Mining Perspective, Southern Methodist University,1999.

[30] Mohammed Shaheen and Bilal Ismaiil, Building Applications Using Java, Java Series 2, Dar mohrat , 2008.

[31] Zhongzhi Shi, General Data Mining Platform-MSMiner, Intelligent Science Research Group, 2003.

[32] Predrag Stanišić and Savo Tomović, Mining Association Rules from Transactional Databases and Apriori Multiple Algorithm, IADIS International Conference, 2008.

[33] Introduction to Data Mining and Knowledge Discovery, Two Crows Corporation, 2005.

[34] William John Teahan, Artificial Intelligence-Agents and Environments, William John Teahan & Ventus Publishing Aps, 2010.

[35] William John Teahan, Artificial Intelligence-Agent BehaviorI, William John Teahan & Ventus Publishing Aps, 2010.

[37] Ian H.Witten, Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, 2005.

[38] X.Wu, V.Kumar, J.R.Quinlan, J.Ghosh, Q. Yang, H. Motoda, G.Mclachlan, A.Ng, B.Liu, P.Yu, Z.Zhou, M.Steinbach, D. Hand, D.Steinberg , Top 10 algorithms in data mining, Knowledge and Information System KAIS, vol.14, no.1, pp.1-37, Springer – Verlag London Limited, 2008.

[39] Xindog Wu and Vipin Kumar, The Top Ten Algorithms in Data Mining, Taylor & Francis Group LLC, 2009.

[40] Osmar R. Zaiane, Principles of knowledge discovery in databases, University of Alberta , 1999.

[41] Mohammed J.Zaki and Limsoon Wong, Data Mining Techniques, Lecture notes series, 2003.

[42] Shicha0 Zhang and Chengqi Zhang, Data Preparation For Data Mining, Applied Artificial Intelligence, vol.17, pp.375-381, Taylor & Francis Group, 2003.

[43] Zijian Zheng, Ron Kohavi and Llew Mason, Real World Performance of Association Rule Algorithms, KDD, 2001.