CHAPTER 3
FRAMEWORK OF THE AUTOMATED DATA MINING SYSTEM AND ROLE OF SOFTWARE AGENTS

This chapter introduces the general framework of the automated data mining system for an object oriented data base. This chapter also explains the role of software agents used in the automated data mining system. It also details the different types of software agents used in this research work.

3.1 Architecture of the Automated Data Mining System

The simple block diagram shown in Figure 3a summarizes the components of the Automated Data Mining System. There are 5 entities in this block diagram and they are explained below.

a) Dataset: Contains object oriented database of \( n \) objects and \( m \) attributes. Dataset from Campus Management System of Karunya University is used for this research.

b) Partitioning: The Object Oriented Dataset is partitioned using the partitioning agent. Vertical partitioning is used in this research work. Partitioning algorithm is explained in detail on Chapter 4.1.

c) Ranking: The partitioned object oriented data set is ranked based on the different ranking factors. With the help of the ranking agent ranking is done using one of the ranking factors to produce a ranked list

d) Selection of Clustering algorithm: The selection of clustering selects a suitable clustering algorithm for the ranked object oriented attributes data set.
RESULT/ OUTPUT

Figure 3a: A block diagram showing an overview of the operations of the experimental data mining system.

The links (arrows) connecting the entities in the block diagram are explained below.

1. The Object Oriented Data set is passed as input into the system.
2. The partitioning employed produces a partitioned object oriented data set.
3. The ranking factors employed by the ranking algorithm produces a ranked list of attributes according to their relevance to the query.
4. The selection of suitable clustering algorithm selects x number of attributes for clustering based on the ranked list.
5. A final clustered data is returned as a result based on the input query.
The framework of the newly developed automated data mining system is represented diagrammatically as follows.

![Diagram of the automated data mining system](image)

**Figure 3b: Framework of the automated data mining system**

In this developed framework an automated data mining system gets the necessary data from the given Object Oriented database. An Object Oriented data set of the Karunya university campus management system is considered for this research work. The user submits the query or gives the input or high level goal or objectives or user’s hint. The user interface agent presented in the data mining system and senses the environment and also analyzes the Object Oriented data set. Based on the submission of the query by the user it classifies them according to the following classification.
It identifies the relationships between the objects, classes, and attributes. It identifies the importance of each of the attributes. The user interface agent also identifies the type of the attributes. After the user interface agent analyzes and classifies the object oriented database based on the user's high level goal or objectives or user's hint the action takes place. The user interface agent identifies the attribute for the action. The analyzed and classified information is communicated to the partitioning agent.

Partitioning agent gets the analyzed information from user interface agent. The Object Oriented data is partitioned vertically with the help of the partitioning processor based on the information given by the user interface agent. The partitioning agent selects the partitioning method that is suitable for the attribute. In addition, the partitioning agent will decide the number of partitions based on the relationship between the attributes, classes, objects, and methods. Once the object oriented data is partitioned ranking agent is called by the partitioning agent.

Ranking is done by Query weight and properties of the attribute type and Similarity between the objects. Ranking agent ranks all the attributes that is partitioned by the partitioning agent and prompts the highest ranked attribute based on the user hints or query. The highest ranked attributes are taken for clustering.

Finally, the Data mining agent lists the possible clustering algorithms for the ranked attribute. The Data Mining agent contains specific clustering
algorithms implemented in house, that may have been installed within the environment. Data mining agent decides which clustering algorithms are appropriate for the ranked data and it selects the appropriate clustering algorithm. Thus, the Data mining agent is responsible for performing the actual data mining activity and generating the results.

Thus necessary knowledge is mined using the clustering algorithm with less user intervention. The detailed analysis of the results of the mined data is explained in Chapter 5.

3.2 Role of Software Agents

Software agents are becoming a more and more important software development technology. The key sign of this trend is the emergence of diverse applications and approaches in many different areas. Software agents are becoming a mainstream technology used widely in many areas including intelligent user interface, industry, Internet and WWW, electronic commerce, business process management, digital library, electronic meeting, entertainment, network management, virtual reality, artificial society, and so on. A software agent is viewed as an autonomous software construction.

Software agents are “autonomous, (preferably) intelligent, collaborative, adaptive computational entities” that have the ability to affect their environment. Software agents aim to help the users with less human intervention to mine the knowledge from the object oriented data. Automated Data Mining System is modeled in terms of Software Agents. In this research work software agents are
used to interact and negotiate with each other. Multiple agents are used in this research work. Each agent is intelligent in its own field and may interact with its human counterpart or behave autonomously.

The software agents used in this research work model reasoning behaviour according to the theoretical Belief Desire Intention (BDI) model of artificial intelligence. Software agents used in this research work are autonomous software components that have explicit goals to achieve or events to handle (desires). To describe how they should achieve these desires, BDI agents are programmed with a set of plans. Each plan describes how to achieve a goal under varying circumstances. Set to work, the agent pursues its given goals (desires), adopting the appropriate plans (intentions) according to its current set of object oriented data (beliefs) about the state of the world.

The software agents used in this research work can exhibit reasoning behaviour under both proactive (goal directed) and reactive (event driven) stimuli. Each agent has:

- a set of beliefs about the world (its data set),
- a set of events that it will respond to,
- a set of goals that it may desire to achieve (either at the request of an external agent as a consequence of an event, or when one or more of its beliefs change), and
- a set of plans that describe how it can handle the goals or events that may arise.
When an agent is instantiated in an automated data mining system, it will wait until it is given a goal (query) to achieve or experiences an event that it must respond to. When such a goal or event arises, it determines what course of action it will take. If the agent already believes that the goal or event has been handled (as may happen when it is asked to do something that it believes has already been achieved), it looks through its plans to find those that are relevant to the request and applicable to the situation. If it has any problems executing this plan, it looks for others that might apply and keeps cycling through its alternatives until it succeeds or all alternatives are exhausted. If it has not happened already it will execute the plan.

In this research work for example when the user submits his goal or query or user hint the developed automated data mining system will take care of partitioning the data, ranking the data and choosing the necessary parameters using the software agents with less human intervention. If the user is using the system frequently the agent believes that the goal has been handled already it executes the plan immediately ie prompts the user about the previous results that has been executed already.

In this research work software agent helps in automating the difficulties in the vertical partitioning of object oriented data, ranking and selection of the attributes, selection of appropriate clustering algorithms and clustering the object oriented data in the automated data mining system. In this research work the Automated Data Mining System is designed using the four types of agents. They
are namely User Interface Agent, Partitioning Agent, Ranking Agent and Data Mining Agents. These agents are autonomous reasoning entities capable of making pro-active decisions while reacting to events in their environment. All the agents used in this research work can communicate and interoperate with other agents through a language called Knowledge Query and Manipulation Language (KQML) or that speaks KQML. The Knowledge Query and Manipulation Language (KQML) is a high level language intended to support interoperability among software agents. The agents used in this research work are detailed as below.

### 3.2.1 User Interface Agents

User interface agents can control, support and provide assistance. The user interface agents are used to monitor the user interactions with the application and can control various aspects of that interaction. It can also observes and monitors the actions taken by the user in the interface, learns new short cuts, and suggests better ways of doing the task. The user interface agent cooperates with the user in accomplishing some task in the application. User interface agents learn typically to better assist its user in four [Mae, 94] ways:

i) By observing and imitating the user (ie learning from the user)  
ii) Through receiving positive and negative feedback from the user (Learning from the user)  
iii) By receiving explicit instructions from the user (learning from the user)  
iv) By asking other agents for advice (ie learning from peers)
Figure 3c: Learning ability of User Interface Agents

The user interface agent interacts with the user in assisting him / her to perform data analysis and data mining activities.

The user interface layer is composed of layered architecture. The user interface agent uses Belief Desire and Intention (BDI) architecture. User interface agent's beliefs are based on sensory input. When presented with an input, an agent reasons about its beliefs to determine what to do. When the user's agent decides on an action, it can carry it out directly if it is within its capabilities, but an action involves other software agents.

In this research a user can provide a general description of the problem at hand in terms of high level goals and objectives, or provide specific details about the data analysis or mining task to be performed as an input. The user interface agent senses the input around the environment with its belief. Then the user interface agent analyzes the object oriented data set. The user interface agent
analyze the number of attributes, number of classes, number of objects, and number of records of an object oriented data set.

Once the user interface agent analyzes the object oriented data set it classifies them based on the following classification. It identifies the relationships between the objects, classes, and attributes. It identifies the importance of each of the attributes. The user interface agent also identifies the type of the attributes.

After the analyzes and classification is carried out by the user interface agent based on the user’s high level goal or objectives or user’s hint the action takes place. The user interface agent identifies the attribute for the action. The analyzed and classified information is communicated to the partitioning agent.

Step 1: Start

Step 2: Get the user needs

Step 3: Identify / get object oriented Data set

Step 4: Analyze the data set
  
  Step 4.1: Input total number of attributes
  
  Step 4.2: Input total number of classes
  
  Step 4.3: Input total number of objects
  
  Step 4.4: Input total number of records

Step 5: Classify the data set
  
  Step 5.1: Identify the relationship between objects, classes, attributes
  
  Step 5.2: Identify the importance of attributes
Step 5.3: Identify the attribute type

Step 6: Classify the action

Step 6.1: Input user hint

Step 6.2: Identify the attribute for the action

Step 6.3: Identify important classes, attributes, methods etc

Step 7: Communicate the analyzed and classified information to partitioning agent.

Step 8: End

**Figure 3d: Developed procedure for User Interface Agent**

The user interface agent is responsible for receiving user specifications and delivering the mined results. It also keeps track of user preferences.

3.2.2 Partitioning Agent

Vertical Partitioning aims in facilitating efficient way of executing next generation database applications by reducing irrelevant instance variable (attribute) access. In this research work a software agent called partitioning agent is used to vertically partition the object oriented data set. This partitioning agent along with the partitioning processor helps in partitioning the data set.

Partitioning agent communicates with the user interface agents and identifies the attributes closely related to each other. Then, with the help of partitioning processor the partitioning agent, partitions the object oriented data. The detailed vertical partitioning algorithm and how the vertical partitions takes place in an object oriented data are discussed in chapter 4.1. The following is the procedure for partitioning agent used in this research work.
Step 1: Start
Step 2: Get the analyzed information from the user interface agent
Step 3: Select the partitioning method and the number of partitions based on the classes, objects, methods and attributes
Step 4: Partition the object oriented data using the vertical partitioning algorithm (discussed in chapter 4.1.3)
Step 5: Call the Ranking Agent
Step 5: End

Figure 3e: Developed procedure for Partitioning Agent

The analyzed information from the user interface agent is communicated to the partitioning agent. The partitioning agent then automatically selects the partitioning method that is suitable for the attribute. In addition, the partitioning agent will decide the number of partitions based on the relationship between the attributes, classes, objects, and methods.

This research work concentrates on non-overlapping vertical partitioning, where there is no overlapping in the attributes of the vertical partitions. Based on the user query or objectives all the attributes that are closely related to each other are partitioned together into one partition. The partitioning agent partitions the closely related attributes that are analyzed by the user interface agents. Once the object oriented data is partitioned the ranking agent is called by the partitioning agent.
3.2.3 Ranking Agent

The success of data mining algorithms relies on their abilities to identify a small subset of relevant attributes. So in this research work once the object oriented data is partitioned by the partitioned agent ranking of the attributes is done with the help of the ranking agent. The attributes in a data set are ranked in order of their relevance for a particular task or query to form a ranked list.

Ranking agent helps in ranking the attributes. Ranking is done by the following factors, *Query weight* and a *Scoring function*. The highest ranked attributes are taken for clustering.

The ranking agent will wait until it is given a goal to achieve or experiences an event that it must respond to. When the ranking agent is faced with such a goal or query arises, it determines what course of ranking action it will take. If the ranking agent already believes that the goal or event has been handled already (as may happen when it is asked to do something that it believes has already been achieved) it retrieves the same ranking method that has been used. It also looks through its other ranking factors to find those that are relevant to the request and applicable to the situation. The ranking agent ranks the attributes. The procedure for ranking agent is described as below:
Step 1: Start

Step 2: Calculate the scoring function for each attribute
   Step 2.1 Classify the types of attributes
   Step 2.2 Assign Score Value for each attributes

Step 3: Calculate Query Weight
   Step 3.1 Usage of attributes
   Step 3.2 Assigning of weightage

Step 4: Identify the highest ranked attributes

Step 5: End

**Figure 3f: Developed procedure for Ranking Agent**

Ranking agent calculates the scoring function for each attribute. It also classifies the types of attributes whether the attributes are numeric, categorical etc. The ranking agent will assign score value for each attributes. Query weight is calculated based on the query. The formulae for query weight and scoring function are detailed in chapter 4.2.

Once the attributes are ranked the ranked attributes are given to a data mining clustering algorithm. The data mining agent used in the automated data mining system will automatically choose the appropriate clustering algorithm for the ranked data.
3.2.4 Data Mining Agent

One of the primary operations in data mining is clustering. An agent called Data Mining Agent is used in the automated data mining system which has specific data mining methods and algorithms. This agent activates and manages data mining algorithms such as cluster analysis algorithms. In this research work more emphasis has given to most well known algorithms in clustering. Most of the widely used clustering algorithms for an object oriented data are used in this research work. The data mining agent contains specific clustering algorithms implemented in house, that may have been installed within the environment.

Data mining agent decides which clustering algorithms are appropriate for the ranked data set and it selects the appropriate clustering algorithm. Thus, the data mining agent is responsible for performing the actual data mining activity and generating the results. The general procedure for the data mining agent is explained below.

Step 1: Start
Step 2: Identify and Select the ranked attributes
Step 3: Categorize the appropriate clustering algorithm suitable for the attributes
Step 4: Cluster data based on the suitable algorithms, k-means, k-medoids, k-modes, Robust clustering algorithm, cactis, orion.
Step 5: Display results
Step 6: End

Figure 3g: Developed procedure for Data Mining Agent
The data mining agent chooses the appropriate clustering algorithms for the ranked attributes ranked by the ranking agent.

Data Clustering is one of the most useful tasks in data mining process for discovering groups and identifying interesting distributions and patterns in the underlying data.

The data mining agent uses the following clustering algorithms for different kind of data. Each algorithm has a different way for representing its clusters. Several clustering methods are identified for datasets with numeric, categorical and for object oriented attributes.

a) Algorithms for Numerical Attribute
   ➢ K- Means Algorithm
   ➢ K- Medoids or PAM: Partitioning Around Mediods
   ➢ CLARA: Clustering Large Applications

b) Algorithms for Categorical Attribute
   ➢ K- Modes Categorical Clustering
   ➢ Robust Clustering Algorithm (ROCK)

c) Algorithms for Object Oriented Data
   ➢ Cactis Algorithm
   ➢ ORION Clustering Method

For all the above said clustering method and corresponding algorithms and the explanation for each of these algorithms are explained in chapter 4.3.1.
In this Chapter, a framework of an automated data mining system is presented which helps in mining the object oriented data with less human intervention. Also, the various software agents developed and used in this research work are discussed. These software agents are autonomous reasoning entities capable of making pro-active decisions while reacting to events in their environment.