CHAPTER 3

PROBLEM DEFINITION AND METHODOLOGY

3.1 OVERVIEW

The research problem is formulated to overcome the limitations mentioned in the previous chapter. At present, several preprocessing techniques for data preparation for Knowledge Discovery, and also several clustering to mining the data are reported in the literature. From the survey, it is inferred that it has many limitations. To overcome the limitations, preprocessing and clustering techniques for Static and Dynamic data is further explored by combining existing techniques with new representation for knowledge extraction. In this chapter, the objectives of the research problem are briefly described. The data mining is modeled for static and dynamic data. Finally, the methodology adapted to meet the needs of the objectives is described.

3.2 OBJECTIVES OF THE RESEARCH WORK

To overcome the limitations inferred from the literature survey, the research problem is formulated with the following objectives:

> To develop an optimal method for attribute correction.
> To identify a best way of treating missing value in the medical data set.
> To develop a clustering algorithm for dynamically changing medical data and as well as for static medical data set.
> To discover the useful knowledge from the Medical data set.
3.3 RESEARCH PROBLEM

The research problem is defined by considering the above mentioned objectives. The problem is briefly explained below. It is understood from the literature, several techniques exist for data mining. The demand for improved accuracy and efficient data mining is also progressively increasing to meet the requirements for Knowledge Discovery. Here preprocessing technique, namely treating missing values, attribute correction using data mining techniques and also clustering algorithm for static and dynamic medical data is also being discussed.

3.3.1 Model for Clustering

Similar to (OHN et al, 2004), clustering is expressed as a set of objects to be clustered is stored in a dataset D defined by a set of attributes $A_1, \ldots, A_m$ with domains $D_1, \ldots, D_m$, respectively. Each object in D is represented by a tuple $t \in D_1 \times \ldots \times D_m$. In terms of the clustering problem, consider two general data types, namely, numeric and categorical. The domains of attributes associated with these two types are called numerical and categorical, respectively. A numerical domain consists of continuous real values. As such, each numerical data object is considered as a point in a multi-dimensional metric space, adopting a distance metric such as the Euclidean or the Mahalanobis measure (Jain and Dubes, 1988). Following the lines of (Huang, 1998), a domain $D_i$ is defined as categorical if it is finite and unordered, e.g., that only a comparison operation is allowed in $D_i$. That is, for any $a, b \in D_i$ either $a = b$ or $a \neq b$. Symbolic data objects as considered in (Gowda and Diday, 1991). Logically, each data object $X$ in the dataset is also represented as a conjunction of attribute, value pairs

$$[A_1 = x_1] \land \ldots \land [A_m = x_m],$$

where $x_i \in D_i$ for $1 \leq i \leq m$. For simplicity, represent $X$ as a tuple

$$(x_1, \ldots, x_m) \in D_1 \times \ldots \times D_m.$$
If all Di’s are categorical domains, then objects in D are called categorical objects. Huang (1997; 1998) also considered the clustering problem for mixed-type data objects where some domains are numeric, while others are categorical.

### 3.3.2 Model for Preprocessing

\[
Y = (y_1, y_2, \ldots, y_p) : \text{complete responses} \tag{3.3}
\]
\[
= (Y_{\text{obs}}; Y_{\text{mis}})
\]
\[
R = (r_1, r_2, \ldots, r_p) : \text{missingness indicators} \tag{3.4}
\]

\(1 = \text{observed}, 0 = \text{missing})

For statistical analysis, the joint distribution of Y and R should be modeled. If factor \(P(Y; R)\) into \(P(R|Y)\) and \(P(Y)\) (i.e. \(P(Y, R) = P(R|Y)P(Y)\)), need the following statistical models:

\[
P(Y, \emptyset) : \text{Complete data model} \tag{3.5}
\]
\[
P(R|Y, \eta) : \text{missingness mechanism} \tag{3.6}
\]

### 3.4 THE GENERAL MODEL FOR KNOWLEDGE DISCOVERY

Figure 3.1 depicts the various phases involved in the discovery of information from data. They are

i) Data Collection

ii) Preprocessing

iii) Data Mining

iv) Knowledge Discovery.
Figure 3.2 shows the data flow of this research work and the three layers of KDD process are also indicated in the diagram.

Figure 3.1 The General Model for Knowledge Discovery in this research work
3.4.1 Data Collection

Data is collected from various sources and is an important aspect of any type of research study. Inaccurate data collection can impact the result of a study or research ultimately lead to invalid results. Data collection methods for impact evaluation vary along a continuum. At the one end of this continuum are quantitative methods and at the other end of the continuum are qualitative methods for data collection.
3.4.2 Data Preprocessing Phase

This phase is a core part of research work, which takes input an Imputated data and output a data set as preprocessed. These steps typically consist of: Data Cleaning (Handling missing values, attribute correction) Data Integration, Data Reducation and Data Discretization.

**Data Cleaning:** Detecting outlier in the database, Handling Missing value using Estimated Maximazation, Attribute correction using data mining techniques, Missing value Imputation methods.

**Data Integration:** It deals with integrating heterogeneous data sources and it is a complex activity that involves reconciliation at various levels - data models, data schema and data instances. In this research work knowledge repository is proposed and also used Jaro–Winkler distance is a measure similarity measure between two strings. This method is used for record linkage First compare the words in the knowledge repository with the given string. If the string does not lexically match, then compare it using JARO–Winkler method.

**Data Discretization:** Discretization is a process that transforms data containing a quantitative attribute so that the attribute in question is replaced by a qualitative attribute. First, discretization divides the value range of the quantitative attribute into a finite number of intervals The mapping function associates all of the quantitative values in a single interval to a single qualitative value. The simplest and efficient discretization method is an unsupervised direct method named equal width discretization [Jiawei Han and Micheline Kamber, 2005] which is a binning methodology. It calculates the maximum and the
minimum for the feature that is being discretized and partitions the range observed in $k$ approximately equal sized intervals.

**Data Reduction:** Data reduction reduces the data set and provides a smaller volume data set, which yields similar results as the complete data sets. Two methods implemented on this research work one are Dimensionality reduction and Numerosity reduction. i) Dimensionality reduction defined as removal of unimportant attributes. II) Numerosity Reduction is fit data into the model. This method can be handled by Parametric Methods. The parameter on which the numerosity reduction has to take place is input from the user. According to the parameter its corresponding values are stored and the remaining data are discarded.

**3.4.3 Data Mining using Clustering**

This phase is another core part of the research, which concentrates on data mining technique using clustering.

Clustering is an automated process to group related records together. Related records are grouped together on the basis of having similar values for attributes. This approach of segmenting the database via clustering analysis is often used as an exploratory technique because it is not necessary for the end-user/analyst to specify ahead of time how records should be related together. In fact, the objective of the analysis is often to discover segments or clusters, and then examines the attributes and values that define the clusters or segments.

**For Static data:**

**Enhanced k-mode algorithm for categorical data using dissimilarity measures:** In this research work, the K-modes algorithm is proposed along with a solution to reduce the initialization problem for the static data set. The proposed
algorithm includes the following steps. Get the input: the number of datasets, number of clustering attributes, the value of K and the dataset as text files are given as input. The K initial models are found out. Depending upon the dissimilarity measure, and the data points are clustered which are close to the initial modes. The initial mode is recalculated and the data points are assigned to the nearest mode. The previous step is repeated until all data points are correctly assigned to their closest mode. The clustered results are then stored in text files.

For dynamic data:

**Enhanced K-means algorithm for dynamic mining:** The proposed algorithm finds the number of clusters on the run based on the cluster quality output. This method works for both the cases, i.e. for a known number of clusters in advance as well as an unknown number of clusters. The user has the flexibility to either fix the number of clusters or the input the minimum number of clusters required. In the former case it works the same as the K-means algorithm. In the latter case the algorithm computes the new clusters by incrementing the cluster counter by one in each iteration until it satisfies the validity of cluster quality threshold. The input for the algorithm is k initial number of clusters (for dynamic clustering) whether the number of clusters is fixed or not and a data set containing n objects. The intra and inter similarity between the cluster is calculated.

**Fast K-means clustering algorithm for dynamic mining:** The proposed algorithm is maintaining the two data structures. The first data structure can be used to store the label of the cluster and the second data structure can be used to store the distance of each data item to the nearest cluster center. This information can be used in the next iteration.
3.4.4 Knowledge Discovery Phase

This phase helps the user to extracts relevant information and factual information from the data. It automatically extracts the desired information from the mined data set. By combining knowledge about the domain helps the user to correlate the information mined from the Medical data.

3.5 EXECUTION ENVIRONMENT

All modules in this research work are implemented by making use of

- JDK 1.7
- Net Beans IDE

3.6 DATASETS USED FOR THE EXPERIMENTS

The datasets that have been used in this research work have been taken from the University of California, Irvine (UCI) machine learning repository. The UCI Machine Learning Repository is a collection of databases, domain theories, and data generators that are used by the machine learning community for the empirical analysis of machine learning algorithms. The data was collected from the Cleveland Clinic Foundation; the Hungarian Institute of Cardiology; the V.A Medical Center and the Zurich University Hospital. The archive was created as an ftp archive in 1987 by David Aha and fellow graduate students at UC Irvine. Since that time, it has been widely used by students, educators, and researchers all over the world as a primary source of machine learning data sets. As an indication of the impact of the archive, it has been cited over 1000 times, making it one of the top 100 most cited “papers” in all of computer science (as such adopted from Frank and Asunetion 2010, Bache and Lichman 2013). The details of the datasets are available in http://archive.ics.uci.edu/ml/about.html.
3.7 PERFORMANCE EVALUATION

F-Measure and Purity are the metrics used to analyze the performance of clustering.

The F-Measure uses the ideas of precision and recall from information retrieval. Each class \( i \) (as given by the class labels of the used benchmark data set) is regarded as the set of \( n_i \) items desired for a query. Each cluster \( j \) (generated by the algorithm) is regarded as the set of \( n_j \) items retrieved for a query, \( n_{ij} \) gives the number of elements of class \( i \) within cluster \( j \). For each class \( i \) and cluster \( j \) precision and recall are then defined as

\[
\text{recall}(i, j) = \frac{n_{ij}}{n_i} \tag{3.1}
\]

\[
\text{precision}(i, j) = \frac{n_{ij}}{n_j} \tag{3.2}
\]

\( n_{ij} \) is the number of members of class \( i \) in cluster \( j \),

\( n_j \) is the number of members of cluster \( j \) and

\( n_i \) is the number of members of class \( i \)

\[
F(i, j) = \frac{2 \cdot \text{precision}(i, j) \cdot \text{recall}(i, j)}{\text{precision}(i, j) + \text{recall}(i, j)} \tag{3.3}
\]

Total F-measure

\[
F^* = \sum_{i=1}^{n} \frac{n_i}{n} \max_{j \in J} \{F(i, j)\} \tag{3.4}
\]
The purity of clusters was used as a measure to test the quality of the clusters. The purity of a cluster is defined as:

\[ P_j = \frac{1}{n_j} \max \left( n^i_j \right) \]  

(3.5)

where \( n^i_j \) is the number of objects in cluster \( j \) with class label \( i \).

The overall purity of the clustering solution is obtained as a weighted sum of the individual cluster purities and is given as

\[ P = \sum_{j=1}^{m} \frac{n_j}{n} P_j \]  

(3.6)

where \( n_j \) is the size of cluster \( j \), \( m \) is the number of clusters, and \( n \) is the number of data points.

According to this measure, a higher value of overall purity indicates a better clustering result, with perfect clustering yielding a value of 1. The similarity measures have been used in Kim et al. (2004) and Guha et al. (2000).

### 3.8 SUMMARY

In this chapter, the objectives of the research problem have been described. The problem definition is modeled for preprocessing and data mining techniques. Also, the general system model for static and dynamic data mining is demonstrated which describes the various modules incorporated in the system, namely, input of the system, algorithms/technique used and output from the system. The details of dataset description, performance evaluation metrics and implementation environment have also been highlighted.