International Journal of Database Theory and Application
Vol.7, No.4 (2014), pp. 99-128
http://dx.doi.org/10.14257/ijdta.2014.7.4.09

A Survey on Pre-processing and Post-processing Techniques in Data Mining

Divya Tomar and Sonali Agarwal
Indian Institute of Information Technology, Allahabad
divyatomar26@gmail.com and sonali@iiita.ac.in

Abstract

Knowledge Discovery in Databases (KDD) covers various processes of exploring useful information from voluminous data. These data may contain several inconsistencies, missing records or irrelevant features, which make the knowledge extraction, a difficult process. So, it is essential to apply pre-processing techniques to these data in order to enhance its quality. Detailed description of data cleaning, imbalanced data handling and dimensionality reduction pre-processing techniques are depicted in this paper. Another important aspect of Knowledge Discovery is to filter, integrate, visualize and evaluate the extracted knowledge. In this paper, several visualization techniques such as scatter plots, parallel co-ordinates and pixel oriented technique are explained. The paper also includes detail descriptions of three visualization tools which are DBMiner, Spotfire and WinViz along with their comparative evaluation on the basis of certain criteria. It also highlights the research opportunities and challenges of Knowledge Discovery process.

Keywords: *Data Preprocessing; Post-processing; Data Cleaning; Feature Selection; Feature Extraction; Data Visualization*

1. Introduction

Knowledge Discovery in Databases (KDD) is the process of exploring valuable, understandable and novel information from large and complex data repositories [1]. Data Mining is a part of KDD process and it performs exploratory analysis and modeling of large data using classification, association, clustering and many other algorithms. KDD process interprets the results obtained from datasets by incorporating prior knowledge. KDD process starts with establishing its goal and ends with the interpretation and evaluation of the discovered knowledge [1-3]. KDD process is iterative in nature and involves following 7 steps as shown in figure 1.

i. *Domain Understanding and set KDD goals:* This is the first and significant step of KDD process. It is essential for the people who perform KDD process to determine the end-user goal and to have good domain knowledge of the background in which KDD process will take place.

ii. *Selecting and creating target dataset:* Having defined the goal, the second step is to select the target data and to create a database on which knowledge discovery process will be executed. This step involves-

- ➢ Check the availability of data.
- ➢ Obtain supplementary essential data.

ISSN: 2005-4270 IJDTA

- Integrate all the data.

This step is very important because the entire knowledge discovery process depends on the available data. Data Mining algorithms learn and explore valuable patterns from this data. Unavailability of important data leads to the failure or wrong interpretation of knowledge. Since the available data is the base of knowledge discovery model so it is essential to obtain important attributes of this data.

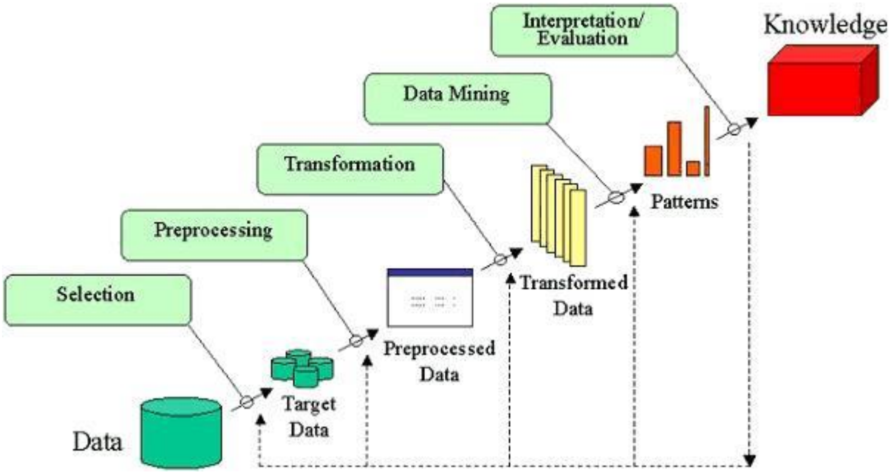

Figure 1. Steps of KDD Process

iii. *Pre-processing:* This step is used to enhance the reliability of the collected data. Pre - processing of data includes data cleaning, for example, handling missing attributes, imbalanced dataset and elimination of noise or outliers. Numerous methods such as filter, imputation and embedded methods are exists to handle missing attributes problem. Filter method discard or eliminate missing attributes from the dataset while imputation based method replace the missing attributes by suitable value. Imbalanced data are handled by either sampling or algorithm modification method which we will discuss in detail in further sections.

iv. *Data Transformation:* Data Transformation process transforms or reproduces the data suitable for Data Mining techniques. Dimension reduction and attributes transformation are some data transformation methods. It is usually specific to the project.

v. *Selection of appropriate Data Mining technique:* This step involves selection of suitable Data Mining techniques such as clustering, association, regression, classification etc. Appropriate selection of Data Mining techniques depend on the goal of the KDD process and also on the previous steps. Data Mining includes two goals: Prediction and Description. Predictive Data Mining includes supervised data mining approaches while Descriptive Data Mining refers to unsupervised approach and also focuses on visualizing the features in the dataset. After determining which Data Mining technique is suitable for KDD process, the next step is to select specific method to discover patterns. For example, if one wants better understandability then Decision Tree is appropriate classification approach while for better precision one can refer to Neural Network or Support Vector Machine. Thus this step is helpful to understand the suitability of a Data Mining technique in particular conditions.

vi. *Evaluation:* This step involves evaluation and interpretation of mined patterns with respect to the goal defined in the first step. This step also determines the usefulness of the proposed model and documents the discovered knowledge for further use.

vii. *Usefulness of the discovered Knowledge:* The knowledge obtained from previous stages is incorporated by the end user in this stage.

It is required to obtain quality and relevant data before applying Data Mining techniques to get useful knowledge. Poor quality data leads to wrong interpretation of knowledge. It is also essential to interpret the discovered knowledge for further decision making. So, pre-processing and post-processing techniques together play significant role in Data Mining. In this paper, we discussed various pre -processing and post-processing techniques of Data Mining in detail.

The paper is divided into five sections as follows: pre-processing techniques such as data cleaning, imbalanced data handling and dimensionality reduction are discussed in Section 2. Section 3 includes various post-processing techniques along with three visualization tools. Section 4 and Section 5 discuss the research opportunities and conclusion respectively.

2. Pre-Processing Techniques

Pre-processing technique includes data cleaning, handling imbalanced dataset and dimensionality reduction such as feature extraction and feature selection.

2.1. Data Cleaning

Quality of data plays important role in information -oriented organizations, where the knowledge is extracted from data. Consistency, completeness, accuracy, validity and timeliness are the important characteristics of quality data. So, i t is important to obtain quality data for knowledge extraction. Data Cleaning is an important step of KDD process in order to recognize any inconsistency and incompleteness in the dataset and to improve its quality [3]. The example of dirty data is shown in Table 1:

Table 1. Example of Dirty Data

S.No.	Dirty Data	Problem
1.	Address=00	Incomplete record
2.	Gender=S	Illegal value

3.	Cus1_name= “Robert S
	Cus2_name= “R. Salva
		Duplicate record

4.	Name= “Robert-09- 1981”26	Multiple values in single column

Dirty data leads to poor interpretation of knowledge. So, Data Cleaning is required to maintain Data Warehousing and deals with identifying and deleting errors and inconsistencies from data to enhance its quality. Several Data Cleaning techniques for handling missing attributes and noisy data are explained as follows:

2.1.1. Methods of Handling Missing Attributes

Incomplete records or missing values produce a challenge to data mining process and can lead to wrong interpretation of knowledge [4]. An instance is called incomplete if there exists at least one missing or unknown value for any attribute. An incomplete instance is represented as:

$$(x_j)_{inc}=(x_{(1)}, \ldots, x_{(p-1)}, ?, x_{(p+1)}, \ldots, x_{(n)}, y)_j \quad (1)$$

where above mentioned instance contains 'n' attributes as xi, y denotes class, and "?" represents missing value for xp attribute. An instance may contain several missing values. Machine Learning (ML) approaches are not designed to deal with missing values and also produce incorrect results if implemented with this drawback. Before applying machine learning approach, it is essential either replace the missing values with some appropriate values or to remove the instances having missing values [4].

Missing values are handled either by replacing it with other values or by removing the incomplete instance. Some ML algorithm such as C4.5 Decision Tree algorithm handles missing values very well [4]. Filter, Imputation and Embedded methods exist to handle missing attributes problem as discussed below:

a. Filter-based Method: Filter based method removes or filter out the instances having incomplete records. This method is suitable only when the numbers of incomplete instances are less. So, their influence on the remaining data is negligible. But when a dataset contains large number of missing records then this method is not a good choice. For large incomplete records, this method induces bias which further leads to wrong interpretation or poor results. Despite this drawback, Filter based approach is one of the most popular and common approach for handling missing values.

b. Imputation Method: Imputation method is based on the assumption that there exist some correlations between the incomplete values and the complete values in the dataset. This approach is more efficient than filter based method as it consider the relationship between complete and incomplete values for handling missing values. This method fills the missing values by a replacement values which is drawn from the available data. Sometimes the estimation depends on the type of data used such as local or global and sometimes it is based on the missing attributes or non -missing attributes. Different types of imputation approaches are discussed below:

Local imputation

This method produces the replacement for each missing records by searching a neighboring records which is similar to it. If any record has missing values, then this method search for the record which is more similar to it and then the missing values are replaced by the values drawn from this neighboring record.

Global imputation based on missing attribute

Global imputation method produces the replacement for missing attribute by analyzing the existing values for that particular attribute. It then replaces the missing attribute by taking mean, median or mode of the existing values of that attribute. This method is simple but reduces the variability in the dataset and weakens the covariance estimation of the data.

Global imputation based on non-missing attribute

This method considers the correlations between missing and non -missing values to find out the replacement for missing attributes. Using these correlations, this method predicts the values for missing attributes. Prediction by regression is one such example of this type of method in which missing attributes is treated as a target attribute while other attributes help to derive or predict this target attribute. But the selection of appropriate regression technique (such as linear or logistic regression) is one of the major problems with this method.

c. Embedded Method: Decision Tree such as C4.5 is able to handle the missing data during training phase [4]. On the basis of number of complete records, it adjusts the gain ratio of each attribute in the training phase. Every incomplete record is scattered among all partitions by using a probability which is measured on the basis of the partition's size. When a test is performed on missing attribute in prediction phase, the instance is propagated again on all available paths. On every edges of Decision Tree, certain weight is mentioned which is based on relative frequency of a value assigned to that edge. CART (Classification and Regression Trees) uses "Surrogate Variable Splitting (SVS)" strategy to replace missing record. For each missing value, it substitutes the value of primary splitter with "surrogate splitter" which is a predictor variable that obtained similar splitting results with the primary splitter [4].

2.1.2. Methods of handling Noisy Data

Noise is an error which occurs randomly in a measured variable. There are several methods to handle noisy data such as binning, clustering and regression. In binning approach, the neighboring data is used to smooth the sorted data values which are arranged into a number of bins [3]. The value in each bin is smooth out by substituting it with the mean or median value of this bin. Bin boundaries, the minimum and maximum value in each bin, are also used to smooth out the bin value. Each bin value is substituted by closest boundary value. Clustering is used to detect the outlier. Clustering technique organizes the similar data points into one group. The data point that lies outside the boundary of this group is an outlier having unusual pattern. Another approach for handling noisy data is regression approach in which data can be smoothed out by fitting into a regression function. Various regression techniques such as linear, multiple or logistic regression are used to determine regression function.

2.2. Handling Imbalanced Dataset

When different classes contain different number of data samples then the dataset is called imbalanced dataset. In such type of dataset, the samples of one class outnumber the data samples of other classes *i.e.*, there is non-uniform or imbalanced distribution of data samples to the classes [5]. In several real world situations, it is very common to obtain data samples in which classes are distribute in non -uniform fashion for example criminal records, fraudulent loan cases and also in the case of genetic data. Such imbalance in data affects the performance of Machine Learning techniques. For example, due to imbalance nature of data, the classifiers become bias towards the class having large number of data samples. Therefore, it is essential to handle class imbalance problem in order to enhance the performance of classifier. In this paper, the class with more data samples is referred as majority class and the class with less number of data samples is termed as minority class. There exist two methods to solve this issue-sampling methods and algorithm adjustment method as discussed below:

2.2.1. Sampling Method

This is one of the simplest method in which the data samples are adjusted in such a way that each class has equal number of data samples. Under-sampling and over-sampling are two methods of adjusting the dataset [6 -7]. In under sampling, the data samples of majority classes are removed or deleted so that each class has same number of data samples. The problem with this method is that the class with more data samples loses its information [8-12]. While in over-sampling method, the minority classes are filled with duplicate entries till all the classes have equal number of data samples. It is observed from the experimental analysis that under-sampling method performed well as compared to over-sampling method because over-sampling method generates unnatural bias for minority classes [11].

2.2.2. Algorithm Adjustment Method

Since the data imbalance problem mainly exist in classifiers, so the second method is to adjust the classifiers in order to deal with this problem. Several methods are proposed by the researchers to deal with this problem. One of the most efficient approaches is to assign different cost to the training samples [13 -17]. Boosting SVM classifiers is also proposed by Wang to handle the data imbalance problem [6 -7].

2.3. Data Transformation

Before applying Data Mining techniques, it is essential to transform the data into forms, suitable for mining process. Data Transformation technique transforms the source data as target data according to transformation criteria [3]. Data Transformation criteria includes:

Smoothing: It is used to remove the noise in the dataset. Smoothing of data can be done by using binning, clustering and regression.

Normalization: This approach is used for scaling the attribute data in such a way that they falls within the same range. Normalization of attribute data can be done either by dividing the attribute value with the mean, standard deviation or maximum value of particular column.

Aggregation: In this, aggregation operations are performed on the data to obtain estimated value based on available values. For example, yearly or monthly sales pattern may be aggregated on the basis of daily sales pattern.

2.4. Dimensionality Reduction

Dimension Reduction is the process of reducing the dimension of a dataset either by selecting significant attributes or by transforming attributes in order to obtain new attributes of reduce dimension. There are three types of dimension reduction problem as given below [18]:

- *Hard Dimension Reduction Problem:* Large dataset (ranging from hundreds to thousands of thousands components) usually requires a strict reduction in the dimension. Principal Component Analysis (PCA) and rough set analysis are commonly used for large dimension reduction [19-20].
- *Soft Dimension Reduction Problem:* When the size of the dataset is not very large then it requires few dimension reduction as compared to High Dimension Reduction problem. Factor analysis is used for soft dimension reduction problem [21].
- In Visualization problem, the relationship within a dataset is visually represented. Projection pursuit and Multidimensional Scaling are the ways of handling visualization problem [22-24].

2.4.1. Category of Dimension Reduction Approaches

Dimension Reduction is of two types -Transformation based or Selection based as shown in Figure 2. Transformation based dimensionality reduction technique is also known as Feature Extraction. In Feature Extraction, the K new features are obtained which are the combinations of N original features. Whereas Feature Selection (FS) is used to select K-best features from N-original features and removes rest of the features. Feature Selection chooses best features and also maintains their original representation. While, in Feature Extraction best features have been chosen only after changing the original representation.

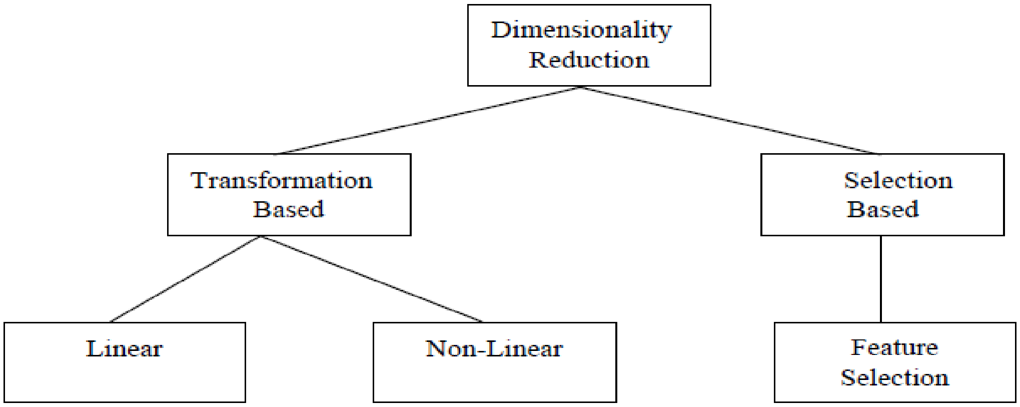

Figure 2. Categories of Dimension Reduction Approach

2.4.1.1. Transformation Based Approach

Transformation based method is divided into two categories- linear and non-linear as shown in Figure 3 [18].

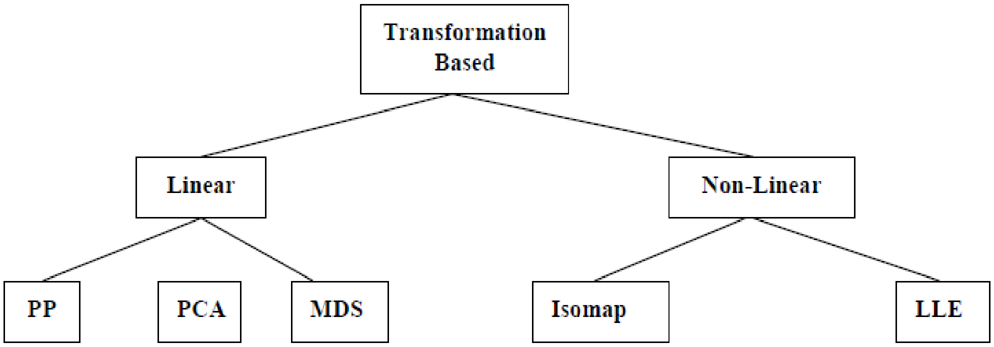

Figure 3. Categories of Transformation based Dimension Reduction Approach

a. Linear Methods: Linear methods embed the data into lower dimension subspace in order to perform dimensionality reduction and also determine the Euclidean structure of the internal relationships of a dataset. Dimensionality reduction using linear method is a very popular approach and includes Principal Component Analysis, Projection Pursuit and Multidimensional Scaling [19, 22-24] as explained below:

Principal Component Analysis: It is one of the common approaches for dimensionality reduction due to its simple theoretical foundation. The objective of the PCA is to obtain reduced dataset and to discover new meaningful variables. PCA transforms the input features in original space into reduced number of features which are uncorrelated with each other [3,19]. These reduced uncorrelated features are also termed as principal components. It constructs a correlation matrix of the data and calculates the eigenvectors of this matrix. Eigenvectors correspond to the largest Eigen -values.

Projection Pursuit (PP): The main purpose of this method is to find the most "interesting" projections in high-dimensional data. A projection is considered to be the most interesting projections if it deviates more from a normal distribution. Each time, when a new projection is found, a reduced data is achieved by deleting the component along with it. The concept behind Projection Pursuit is to determine the projections from multi-dimensional space to lower dimensional space in order to identify the details about the structure of dataset. Although this method works well with irrelevant or noisy variables but still it requires high computational time [22-23].

Multi-Dimensional Scaling (MDS): MDS is a collection of techniques that use the similarities of data points while transforming the data from multi -dimensional space to lower-dimensional space [24]. It preserves the pair-wise distance between data points as much as possible. The pair-wise distance of x_i and x_j in N-dimensional space is calculated by Euclidean metric as:

$$dist(x_i, x_j) = \sqrt{\sum_{p=1}^{N} (x_{ip} - x_{jp})^2} \quad (2)$$

The performance of MDS is may be evaluated while measuring the deviation present in pair-wise distance in high and low dimensional space also known as stress. MDS uses two type of stress function given below:

$$Raw\_stress(Y) = \sum_{ij} (dist(x_i, x_j) - dist(y_i, y_j))^2 \quad (3)$$

$$Sammon_{cost}(Y) = \frac{1}{\sum_{ij} dist(x_i, x_j)} \sum_{i \neq j} \frac{(dist(x_i, x_j) - dist(y_i, y_j))^2}{dist(x_i, x_j)} \quad (4)$$

Where x_i , x_j and y_i ,y_j are data points in high and low dimensional space. Sammon cost function emphasizes on keeping distances that were originally small. Stress function can be optimized by using Pseudo Newton method or conjugate gradient method. MDS is mostly used in fMRI analysis for visualizing data.

b. Non-Linear Methods: Linear transformation based methods are not suitable for the dataset containing non-linear relationship between them. Here we discussed two non - linear transformation methods as:

Kernel Principal Component Analysis (KPCA): It is a combination of PCA and the kernel trick for the dimensionality reduction of non-linear dataset [25]. PCA computes the covariance matrix of the m×n matrix A as:

$$C = \frac{1}{m} \sum_{i=1}^{m} x_i x_i^T \quad (5)$$

Then it maps the data onto the first k-eigenvectors of co-variance matrix. While in KPCA, first kernel trick is used to transform the data into higher dimensional space and then covariance matrix is computed of the data as follows:

$$C = \frac{1}{m} \sum_{i=1}^{m} \emptyset(x_i) \emptyset(x_i)^T \quad (6)$$

where $\emptyset(x_i)$ is any kernel function. It then maps the transformed data onto the first k-eigenvectors of co-variance matrix. The main problem of this technique is the selection of suitable kernel function.

Isomap: MDS considers only Euclidean distances between data points. It does not take into account the distribution of neighboring data points. Isomap, an extension of MDS, preserves the pair-wise geodesic distances between data points [18, 25 -26]. The distance between data points over manifold is considered as geodesic distance and it is calculated by constructing a neighborhood graph. In this graph, every data point is connected with its k-nearest neighbors. The next step is to compute the shortest distance between data points by using Dijkastra's and Flouyd's Warshall shortest path algorithm. The shortest path between data points provides a good estimation of geodesic distance between them. On the basis of geodesic distance between points, a pair -wise geodesic distance matrix is constructed and then MDS transforms it from higher to lower dimensional space by using geodesic distance matrix. Isomap does not work well for non -convex manifold. It also suffers from "holes” in manifold which can be handled by destroying manifolds with holes. Another major problem with Isomap is that it is topological instable. Sometimes, Isomap may establish invalid connections with its neighborhood graph which results poor performance [18]. Apart from these disadvantages, it is widely used in the visualization of bio-medical data and in wood inspection.

Locally Linear Embedding (LLE): It is a local technique of transforming the data from higher dimensional space to lower dimensional space. Similar to Isomap, it generates a graph representation of the data points. LLE maintains the local properties of the data due to which it is less sensitive to short circuiting as compared to Isomap. Only less numbers of properties are affected if short circuiting occurs. It is also useful for non - convex manifold. LLE considers each and every point one by one along with their nearest neighbors and calculates their weights. These weights are useful to obtain each data point from its neighbors by combining them linearly [27]. It uses an eigenvector based optimization approach and transforms the data points into lower dimensional space in which each point in lower dimension also described with the same linear combination of its neighbor. Since each point is described by its neighbors so it can be reconstructed by using weight matrix of its neighbor. The reconstruction error is obtained by the following formulation:

$$E(W) = \sum_i \left| X_i - \sum_j W_{ij} X_j \right|^2 \tag{7}$$

Where W_{ij} is a weight matrix and X_j are neighbor of point X_i. The above cost function is minimized by using two constraints-

- Each data point X_i is reconstructed only with the help of its neighbor. If any point X_j is not a neighbor of it, then the value of corresponding weight W_{ij} should be zero.
- The total value of the weight matrix in every row is equal to 1.

$$\sum_j W_{ij} = 1 \tag{8}$$

The goal of this algorithm is to reduce the dimension of the data points in such a way that the weight W_{ij}, which is used to reconstruct the i-th data point in original dimensional space, will be used to reconstruct the same data point in lower dimensional space. The visualization ability of LLE is very poor and sometime it is not able to visualize the simple synthetic biomedical datasets. When the target dimensionality is very low, then it collapses large portion of data onto a single point.

2.4.1.2. Feature Selection (FS)

In many real world applications, FS is necessary due to the presence of irrelevant, noisy or ambiguous features [4]. The main purpose of FS approach is to select a minimal and relevant feature subset for a given dataset and maintaining its original representation. FS not only reduces the dimensionality of data but also enhance the performance of a classifier. A feature is said to be relevant, if the class is conditionally dependent on it i.e., if the feature is helpful in predicting class attribute. Another important criterion to check the usefulness of feature is tested on the basis of its redundancy where a feature is highly associated with other features [28]. A good feature subset includes those features which are highly correlated with the class attribute or decisions function but are uncorrelated with each other [4]. So, the task of FS is to search for best possible feature subset depending on the problem to be solved. Generally, FS uses random or heuristic search strategies for the selection of optimal feature subset. There are four procedures in Feature Selection approach as indicated in the following figure:

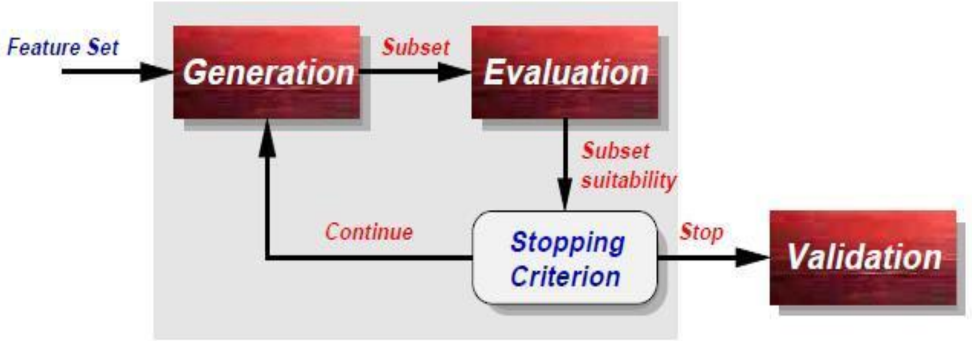

Figure 4. Feature Selection

Generation step of FS approach produces a feature subset for evaluation step. There are 4 possible cases of generating feature subset such as all features, no feature, a random feature subset or selected feature subset [29-30]. If there are all feature in the feature subset then it iteratively removes the feature from it and then evaluate while for second case when there is no feature, it iteratively add features to feature subset for evaluation. For last case, either features are produced randomly or added or removed iteratively [31]. Evaluation step is necessary to evaluate the appropriateness of the identified feature subset after generation step and then compares it with earlier generated feature subset and replacing it if found to be better. Feature subset may be evaluated using several evaluation criteria such as accuracy, error *etc*. A stopping criterion is used to find out the termination cases for feature selection procedure. There are various stopping criterion for example, for fixed number of selected feature or when an optimal feature subset is obtained. After reaching a stopping criterion, the feature subset may be validated for further use. The objective of feature selection technique is 3-fold as:

- Feature Selection reduces the dimension of the original dataset and lower dimension improves the accuracy of a classifier. An optimal feature subset can also enhance the performance of a simple learning algorithm.
- Feature selection reduces the overall computational cost of a learning algorithm.
- Provides better insight.

Characteristics of Feature Selection Algorithm:

In Feature selection, there are four basic issues which affect the search of optimal Feature subset as:

a) ***Starting Point:*** Starting point is a point from where the search for an optimal feature subset begins. Selection of this point in the feature subset space is one of the important tasks that affect the search. There are various methods of selecting starting point as- either starts with no feature and then successively adds features to it or starts with all features and then removes features from it. The first method is known as Forward Selection and second one is Backward Elimination method.

b) ***Search Strategy:*** Mainly Feature Selection uses two search strategies as- Exhaustive and Heuristic search strategy. Heuristic search strategy is more effective and feasible and can produce good results as compared to Exhaustive search approach.

c) ***Evaluation Strategy:*** Evaluation of the feature subset is one of the most important factors in Feature Selection. Different Feature Selection techniques such as Filter and Wrapper adopt different evaluation strategy. Filter method evaluates the feature subset based on the general characteristics of the data. While Wrapper evaluate the selected feature subset on the basis of performance of the learning algorithm.

d) ***Stopping Criterion:*** A Feature Selector must able to decide when to stop searching for the feature subset in the feature space. The stopping criterion of Feature Selection depends on the evaluation strategy. It might stop adding or removing features when it does not produce better results as compared to current one. Another important consideration of deciding stopping criterion is to continue producing feature subset until reaching the opposite end of the search space and then select the best one.

The features are selected by using following two ways:

i) Feature Ranking: The feature ranking approach ranked the features on the basis of certain criteria fixed as per their relevance and then it selects top 'n' ranked features. These top ranked features may be generated automatically or specified by the user. This method ranks the features using several functions such as Euclidean Distance, Correlation Method or Information based criterion. Although the procedure works well with this method but since it considers features separately from each other causes following problems:

- Sometimes any individual feature seems irrelevant and can be discarded but while considering this in combination with other feature may become useful.

- Sometimes feature seems extremely relevant individually but become redundant during analysis.

Feature ranking is not suitable when there exists some correlation between features and when target function is predictive using combined set of features. Feature ranking is suitable for microarray analysis when genes are analyzed in order to differentiate between healthy and sick patients.

ii) Feature Subset Selection: This approach selects a subset of features based on evaluation criteria. On the basis of evaluation approach, Feature selection is of three types- Filter, Wrapper and Embedded Feature Selection approach.

A Filter Feature Selection approach is independent of the learning algorithm. Filter approach discard irrelevant features in problem domain before induction. It first selects the feature subset and then apply learning algorithm on this subset. Subset selection is purely independent of learning algorithm due to which this technique is applicable to any domain [28]. For a given dataset, it generates a relevance index for each feature that determines the relevance or importance of each feature for any given task. A relevance index is also known as feature selection metric and is calculated on the basis of distance, correlation or information. A Wrapper approach is dependent on the learning algorithm. In this approach the suitability of subset is evaluated on the basis of classification accuracy. Wrapper Feature Selection approach is a good choice of feature selection and may produce better results but its computational cost is high and cannot work well with very high dimensional dataset. An Embedded method is specific to given machine learning algorithm and includes feature selection in the learning phase of classifier. Embedded Feature Selection approach is faster than the Wrapper method but suffers from over-fitting problem. Figure 5 and 6 show the working of filter and wrapper feature selection approaches.

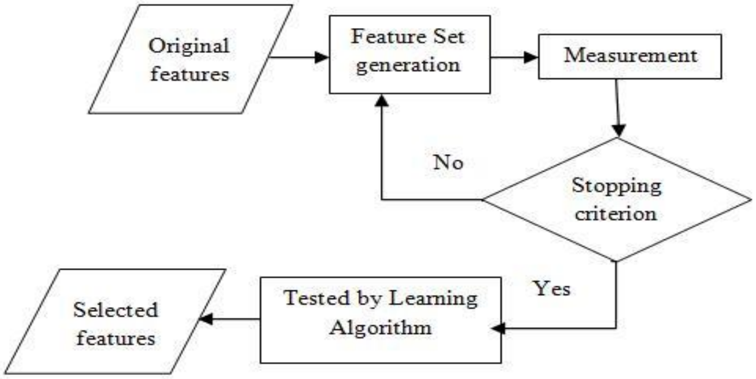

Figure 5. Filter Feature Selection Approach

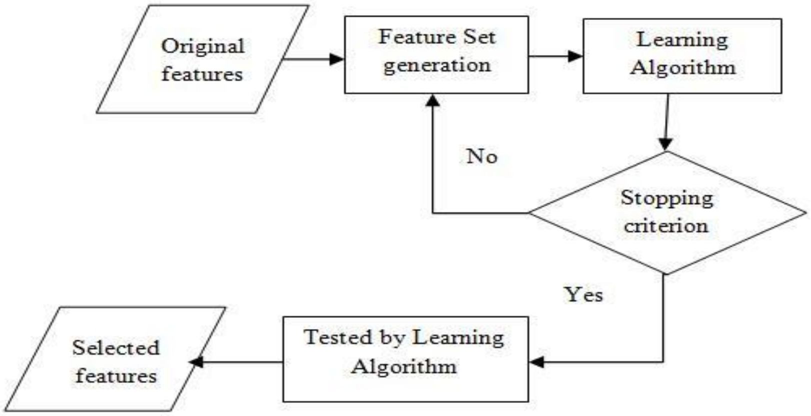

Figure 6. Wrapper Feature Selection Approach

2.4.1.2.1. Feature Selection Techniques

a. Filter Method: This method first selects the feature subset and then apply learning algorithm on this subset. There are several Filter based feature selection techniques which are discussed below and for each techniques we take a common dataset of 2-class to clearly understand the working of each techniques.

Table 2. Example Two-class Dataset

Object	A	b	c	d	e	f	output
0	0	1	1	1	0	0	1
1	1	0	1	1	0	0	1
2	1	1	0	0	0	0	1
3	1	1	0	1	0	1	1
4	1	1	1	0	1	0	1
5	0	0	1	1	0	0	1
6	0	0	0	1	1	1	0
7	0	0	1	0	0	1	0
8	1	0	0	0	1	1	0
9	1	0	1	0	0	1	0
10	0	1	0	0	0	1	0
11	0	1	1	0	1	1	0
12	0	1	1	1	1	0	0

Relief Algorithm: In this Filter based FS approach, each feature is assigned with a relevance weighting that play significant role in the prediction of class labels. The working of Relief algorithm is given below: First it specifies the number of sampled object as first threshold which are used for calculating the weight for each feature [18, 32]. It then randomly draws an object from a dataset and on the basis of distance formula its nearMiss and nearHit are calculated. nearMiss of the randomly drawn object ‘x’ are the nearest objects with different class labels and nearHit are nearest objects with the similar class labels. The distance between two objects x and o are calculated as:

$$dist(o,x) = \sum_{i=1}^{|C|} diff\ (o_i, x_i) \tag{9}$$

Where

$$diff\ (o_i, x_i) = \begin{cases} 1, & o_i \neq x_i \\ 0, & o_i = x_i \end{cases} \tag{10}$$

The Relief algorithm is given below:

RELIEF(D, c, n, ε).
D : Dataset, ε: threshold value, c: # of conditional features, n:# of iterations

```
(1)    R← {}
(2)    $\forall W_a, W_a \leftarrow 0$
(3)    $for\ i = 1, \ldots, n$
```

```
(4)  Randomly choose an object x from dataset D.
(5)  Obtain nearHit and nearMiss of this object.
(6)  For k=1,…,c
(7)      $W_k \leftarrow W_k - \frac{diff(x_k, nearHit_j)}{n} + \frac{diff(x_k, nearMiss_j)}{n}$
(8)  $For\ k = 1, \dots, c$
(9)      If $W_k \geq \varepsilon; R \leftarrow R \cup \{k\}$
(10) Return R.
```

A threshold value, which is specified by the users, determines the relevancy of the selected features. This method is useful to select relevant features but ineffective to discard redundant features which are highly correlated with each other. Relief works on the given examples as follows. First of all, an object is selected randomly, suppose we select object 0, then the next step is to determine its nearHit and nearMiss. Object 5 is its nearHit as its distance is '1' and object 12 is nearMiss as its distance is '3' using above formula. The value of weight parameter is updated for each feature according to their difference i.e., using nearHit and nearMiss and the process is repeated for given number of iterations. The features are inserted into final features subset when their weights exceed the desired level, ε. For 100 iterations and for ε=0, Relief generates the final subset {a; d; e; f}. In this way, this method ranked the features according to their weights. This method is not work well for insufficient number of instances and there is no general criterion for choosing the sample size.

FOCUS Algorithm: Focus uses Breadth First Search to determine a minimal feature subset [33]. It generates all subsets of the given size (initially start with 1) and test each of them to find inconsistency. Particular subset is deleted for any inconsistency. This process is repeated until the algorithm obtains a consistent subset or all possible subset have been checked. This algorithm is sensitive to noise and not suitable for multi-dimensional dataset. Focus algorithm is given below:

FOCUS (D, c).
D: Dataset and c: # of conditional features.

```
(1)  R← {}

(2)  For i=1,…, c
(3)      For each subset L of size i
(4)          Find out Consistency between subset L and dataset D.
(5)          If Consistency==true
(6)              $R \leftarrow L$
(7)              Return R
(8)          Else Continue.
```

For given dataset, it first measures the consistency of all subsets {a}, {b}, {c},{d},{e} and {f} with initial size of subset is 1. It is found that there is no effect of considering these subsets and dataset could not be reduced. For example, object 0 and 12 conflicts with each other for subset {f}. Again all subset of size 2 ({a,b};{a,c};{a,d} etc.) are evaluated and no appropriate subset is found. In the same way, the process is repeated until the subset {a,d,f} is selected which is a minimal subset in terms of consistency criterion.

Las Vegas Feature Selection Algorithm (LVF): LVF, based on probabilistic search approach, randomly select the feature subset and uses consistency evaluation measure [18, 34]. It starts with the selection of best feature subset and performs the feature selection task by assuming it as entire conditional feature set. Then again a feature subset is selected randomly and its cardinality and inconsistency rate is compared with the current best feature subset. The randomly selected feature subset is considered new best one if it has smaller cardinality and having inconsistency rate less as compared to a specified threshold value ε . To obtain inconsistency rate, first there is a need to calculate the inconsistency count. Inconsistency count is obtained by using following formula:

$$IC = \sum IO - MFC \qquad (11)$$

Where IC=Inconsistency Count
IO=Inconsistent objects
MFC= # of objects with most frequent class label.

Inconsistency Rate (IR) is obtained as follows:

$$IR = \frac{\sum IC}{n} \qquad (12)$$

Where ‘n’ is the number of objects. It also works well in the presence of noise. Since, it does not take prior knowledge into consideration, so this method may take more time than heuristic search approach to find the solution. LVF algorithm is given below:

LVF(D,c, n, ε)
D: Dataset, c: set of conditional features; n: # of iterations of the algorithm; ε: consistency threshold.

```
R
For i=1,..., n
Obtain random feature subset and store in S.
If $|S| \le |R|$
  Evaluate the inconsistency between S and Dataset D and compare it with threshold
  value. Inconsistency(S,D)$\le \varepsilon$.
If $|S| < |R|$
      $R \leftarrow S$; output R.
    Else $R \leftarrow R \cup S$
Return R.
```

Correlation based Feature Selection (CFS): As the name indicates, CFS filter feature selection approach considers the correlation of feature with target feature and selects only those features which show a strong correlation with the target feature and weak correlation with each other [35-36]. Correlation coefficient between two features X_i and X_j are calculated as:

$$Correlation(X_i, X_j) = \frac{E[(X_i - \mu_{Xi})(X_j - \mu_{Xj})]}{\sigma_{Xi}\sigma_{Xj}} \qquad (13)$$

where σ and μ represent standard deviation and expected values respectively. Correlation coefficient indicates how strongly two features are related or associated with each other i.e., when the value of one feature is able to predict the value of another feature then they said to be strongly correlated with each other. Correlation can be estimated from training samples as:

$$r_{x(i),x(j)} = \frac{\sum_{k=1}^{m}(x_k^i - \bar{x}^i)(x_k^j - \bar{x}^j)}{(m-1)S_{x(i)}S_{x(j)}} \tag{14}$$

Where $S_{x(i)}$and $S_{x(j)}$ indicate standard deviations of training samples, $\bar{x}^i$and $\bar{x}^j$ are the mean value of sample and x_k^i and x_k^j represent the value set of features X_i and X_j correspondingly. Above measurement value is used to rank the features according to their individual association with the target feature. A feature set is optimal only when it shows strong correlation with target feature and weak correlation with each other. So, high rank is given to the feature that satisfies strong correlation criteria. Using this condition, the merit of a subset of features is given as:

$$M_F = \frac{k\bar{r}_{cf}}{\sqrt{k+k(k-1)\bar{r}_{ff}}} \tag{15}$$

where k denotes number of features, r is sample correlation coefficient, c is class and f is predictive feature. $\bar{r}_{cf}$ and $\bar{r}_{ff}$represent average value of feature-class correlation and feature-feature correlation respectively.

b. Wrapper Method: Wrapper feature selection approach is dependent on the learning algorithm and it identifies optimal feature subset on the basis of particular learning algorithm [18, 28 and 37]. There are mainly three steps of wrapper methodology:

- a generation procedure
- an evaluation procedure
- a validation procedure

Generation procedure generates or selects a candidate feature subset from the original feature space. Evaluation procedure evaluates the performance of the learning algorithm by using candidate feature subset. So, in this way the learning algorithm guide the search for feature subset. The validation procedure checks the suitability of the candidate feature subset by comparing it with other feature selection and generation method pairs. The purpose of validation procedure is to recognize most suitable selection method for given dataset and learning algorithm.

The search strategy in wrapper method either removes or adds features into candidate feature subset and finds an optimal feature subset that maximize the performance of learning algorithm [28]. For example, in case of a classifier, the optimal feature subset maximizes its accuracy. Two common search strategies used in Wrapper methods are given below:

- *Forward Selection:* At the beginning it assumes an empty feature set and then it iteratively choose and add features one by one. In each iteration, the performance of learning algorithm is evaluated by using generated feature subset. This search continues until adding new feature improves the performance of learning algorithm and stops when there is no improvement in the performance of learning algorithm with respect to the current feature subset.

- *Backward Elimination:* It starts with a set of all features and then iteratively removes one feature in each iteration. In each step, the feature is removed only when removal of it enhance the performance of learning algorithm. The search stops when removal of a feature degrades the performance.

Both Backward Elimination and Forward Selection strategies have some advantages and shortcomings. Backward Elimination does not work well when initial numbers of features are very large. In this situation, it takes more time and becomes infeasible whereas Forward Selection is faster when there is a need to select small number of features. On the other hand, Backward Elimination evaluates the contribution of a feature with rest of the potential features. While the added feature using Forward Selection strategy may become useless after some time.

c. ***Embedded Method:*** In this FS method the learning part and the feature selection part is carried out together. Decision Tree classification approach can be considered to be an embedded method in which the tree construction and the attributes selection are interleaved. In each iteration of the tree construction, the attribute selection is usually done by a simple filter method. Another example of embedded feature selection method is L1-SVM [18].

d. ***Hybrid Method:*** Filter method is useful to rank features according to their importance but the main problem is to decide the stopping criterion. Wrapper method is not suitable for large dataset due to exhaustive search. Since Filter is a less computationally expensive feature selection approach and wrapper is more accurate, so Hybrid approach combines both, the advantages of filter and wrapper methods [28, 37]. The working of Hybrid feature selection approach is shown in Figure 7.

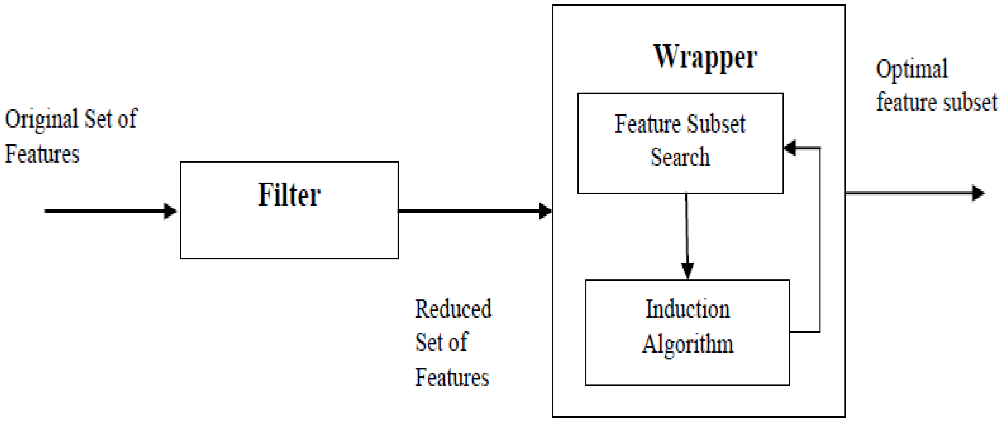

Figure 7. Hybrid Feature Selection Approach

In the first phase of hybrid approach, first the Filter approach is used to remove the redundant and irrelevant features from the data set and then Wrapper method is used to obtain optimal feature subset from relevant features set. The results obtained from literature review reported that Hybrid feature selection approach has more computational speed and predictive accuracy as compared to filter and wrapper approaches [28, 37]. Table 3 discusses the advantage and disadvantage of various feature selection approaches:

Table 3. Comparison of Feature Selection Approaches

FS Method	Advantages	Disadvantages	Examples
Filter Method	Fast and scalable. Independent of Learning Algorithm. Better Computational Complexity than wrapper method.	Not Accurate. Ignores Interaction with the classifier.	Correlation based Feature Selection.
Wrapper Method	Simple. Dependent of learning Algorithm. More Accurate.	Computationally Complex. Not suitable for large dataset. Risk of Over-fitting.	Sequential Feature Selection. Sequential Backward Elimination.
Embedded Method	Interacts with learning algorithm. Better Computationally Complexity than wrapper methods.	Classifier dependent Selection.	Decision Tree. Weighted Naive Bayes. Feature Selection using weight vector of SVM.
Hybrid Method	More accurate than filter and wrapper methods. Better Computationally Complexity.	The selection of suitable filter and wrapper FS technique is the main problem.	Filter + Wrapper + any Learning algorithm

2.4.1.2.2. Applications of Feature Selection

Feature Selection has wide applicability in real world as given below:

- ***Image Recognition System:*** FS are widely used in image recognition system in order to optimize the classification performance [38-39]. FS not only enhance the computation process but also improve the accuracy of learning algorithm. For an instance, initially the accuracy of skin tumor recognition system is noted between 65% to 85 % but with the application of FS, the recognition system produces classification accuracies above 95% [40].
- ***Bio-informatics:*** Gene expression microarrays is one of the fast growing technology that helps to analyze the expression levels of thousands genes with the help of machine learning techniques. A classifier differentiates between cancerous and non-cancerous cells on the basis of patient gene expression. Since gene expression data contains large number of features so it complex the task of classification [41-42]. Feature Selection approach not only help to reduce the size of dataset in terms of reduced, relevant feature subset but also enhance the accuracy of a classifier. QSAR is another application of bio-informatics which obtains the hypotheses regarding chemical features of molecules with their molecular activity [43]. Feature Selection is also used for splice site prediction and detects the junctions between coding and non-coding regions of DNA [44]. Feature Selection is also very useful for disease diagnosis. Healthcare data contains large number of features which are difficult to analyze by a learning algorithm. Sometimes, most of the features are irrelevant which produce wrong interpretation about any particular disease. So, Feature Selection is helpful to construct an effective disease diagnosis system with the help of extracted significant

features. Predictive accuracy of disease diagnosis and software prediction model is also increased with the application of Feature Selection [45-49].

- ***Clustering:*** It is a way of grouping on the basis of certain unexpected characteristics and applied as document clustering, data clustering and clustering of nodes. In all such clustering approaches high dimensionality of the feature space ruins the performance of clustering algorithm [50]. So it is essential to reduce the size of feature space to enhance the performance of clustering.
- ***Text Categorization:*** Documents can be viewed as a collection of words in text categorization. Each document is analyzed using their extracted keywords and rated on the basis of frequency of occurrence. Dimensionality reduction becomes essential as the size of extracted keywords is of the order of tens of thousands. Bookmark and web page categorization are the recent applications of Feature Selection approach [51-52].
- ***Rule Induction:*** If-then rules representation is one of the most common approaches of representing knowledge in human readable form. A feature selection approach is required to reduce the complexity of generated rules and to speed up the process of rule generation. The selection procedure not only reduces the dimensionality of large feature set but also removes the redundancy from it.

3. Post Processing Techniques

It is essential to visualize the extracted knowledge in such a way that user can interpret the knowledge easily. KDD post processing includes following techniques:

3.1. Knowledge Filtering

The two methods of knowledge filtering are- rule truncation and post pruning which is very commonly used with Decision Tree and decision rules. In post processing phase, the main objective is to extract meaningful results to perform certain decision making step. Sometimes, for example decision rules, the classifier generate multiple rules with certain degree of redundancy which is difficult to translate in terms of useful decision making actions. Rule truncation is a post processing method applied in decision rules in which we truncate certain rules to improve the performance of a classifier. Similarly in decision tree, unusual growth of a tree may lead to confusing knowledge interpretation so decision rules may be applied to shrink the tree for better understanding purpose.

3.2. Evaluation

It is essential to evaluate the performance of the system. Classification accuracy, error, computational complexity etc. are used to evaluate the classifier model. Confusion Matrix is used to evaluate the performance of both classification and clustering approaches. Confusion matrix highlights the details of actual/predicted class as shown in Table 4:

Table 4. Confusion Matrix

Actual Class ↓	Predicted Class →	
	Class 1	Class 2
Class 1	True Negative (TN)	False Positive (FP)
Class 2	False Negative (FN)	True Positive (TP)

Table 5 shows the evaluation parameters for clustering and classification:

Table 5. Performance Evaluation Parameters

Clustering Evaluation Parameters	Classification Evaluation Parameters
Davies-Bouldin Index DB=$\frac{1}{n}\sum_{i=1}^{n} max_{i\neq j}(\frac{\sigma_i+\sigma_j}{d(c_i,c_j)})$	Accuracy=$\frac{TP+TN}{TP+TN+FP+FN}$
Dunn Index D=$\min_{1\leq i\leq n}\{\min_{1\leq j\leq n,i\neq j}\{\frac{d(i,j)}{\max_{1\leq k\leq n} d'_k}\}\}$	Recall/Sensitivity=$\frac{TP}{TP+FN}$
Silhouette Coefficient compares the average distance of elements within the same group with the average distance of elements of other groups.	Specificity=$\frac{TN}{TN+FP}$
Rand Index RI=$\frac{TP+TN}{TP+TN+FP+FN}$	Geometric Mean=$\sqrt{Sensitivity * Specificity}$

Precision=$\frac{TP}{TP+FP}$
Recall=$\frac{TP}{TP+FN}$
F-measure=$\frac{2*Recall*Precision}{Precision+Recall}$
	Precision=$\frac{TP}{TP+FP}$

Jaccard Index J=$\frac{TP}{TP+FP+FN}$	Root Mean Squared Error RMSE=$\sqrt{(\sum_{i=1}^{n}(f_i - y_i)\ /n)}$
Fowlkes-Mallows Index FM=$\sqrt{\frac{TP}{TP+FP}.\frac{TP}{TP+FN}}$	Mean Absolute Error MAE=$\frac{1}{n}\sum_{i=1}^{n}\lvert f_i - y_i\rvert = \frac{1}{n}\sum_{i=1}^{n}\lvert e_i\rvert$

3.3. Information Visualizatione

Data Mining is useful for extracting the knowledge from large databases. After extraction, it is important to visualize the extracted knowledge in such form so that user can gain insight into data for better decision making [53]. Various techniques are available for information visualization. Some of them are discussed below:

a. Scatter Plot Matrix Technique: Scatter Plots are organized in matrix form and use Cartesian co-ordinates to plot data points [53-55]. The relationship between two variables, also known as correlation, is represented by scatter plots. The correlation between two variables may be positive or negative. If two variables show high correlation with each other, then data points make a straight line in scatter plot. If the data points are distributed uniformly in the scatter plot, then the correlation between two variables is low or zero. High correlation may be positive or negative depending on the relationship between variables. If the value of one variable increases with the increment of the value of another variable and if data points are represented by a straight line in scatter plot, then the correlation is said to be high positive correlation. If the value of one variable decreases with the increment of the value of another variable and data points make a straight line then correlation is called high negative correlation. Figure 8 indicates the positive and negative correlation between two variables:

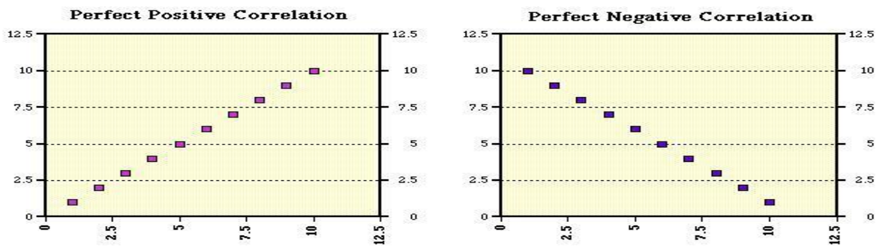

Figure 8. Perfect Positive and Negative Correlation between Two Variables

Scatter plot provides theoretical foundation to many commercial visualization software tools such as DBMiner, Xgobi and Spotfire [56]. Scatter Plot is not a suitable approach for the visualization of large variables and further leads to confusion [57].

b. Parallel Co-ordinates: This technique maps a multi-dimensional point onto a number of parallel axes. Initially in this method coordinates start mapping with one axis and then gradually more axes may be lined up as per requirement. A line is used to connect the individual coordinate mappings. This method may be extended to n-dimensions but there is a practical limit which depends on the screen display area [53, 58]. Figure 8 shows the plot by parallel co-ordinates method for single instance of n-dimension. We can map many instances onto the same set of axes and structures in data are obtained by the patterns formed by polygonal lines.

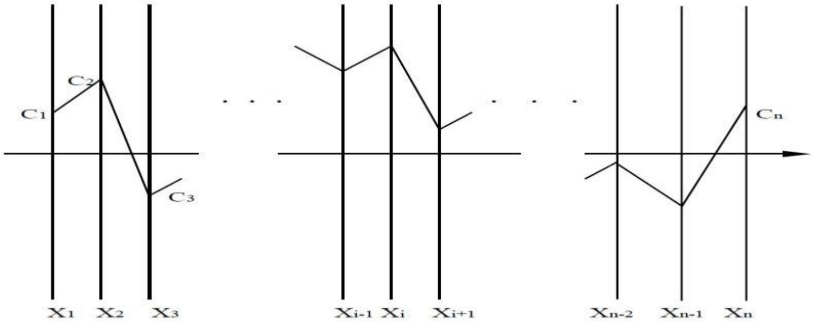

Figure 9. Parallel Axes

This visualization technique is used in air traffic control system, computer vision, robotics *etc*. [60]. Keim *et al.*, along with Lee and Ong have also included this visualization data mining techniques in their software VisDB and WinViz respectively [59-60]. The main advantage of using this technique is that it is suitable for the visualization of multi-dimensional dataset. The only problem with this technique is representation of multiple points by same parallel axes which sometimes difficult to visualize due to overlapping. Keim *et al.*, performed an analysis of the visualization technique and concluded that around 1000 data points can be visualized on the screen at the same time.

c. Pixel Oriented Technique: In this technique each individual pixel in the screen is used to display an attribute value for some data sample. On the basis of attribute value, a color is assigned to the pixel. Since there are large number of data samples in the screen so we can easily represent as many attributes as pixels in the screen [61]. Therefore, this technique is suitable for the visualization of large data samples. The application of pixel oriented techniques is not only to demonstrate all possible pixel arrangement on the screen but also show the display patterns in terms of number of windows as per dataset dimension.

3.3.1. Information Visualization Tools

There are several tools available for information visualization. DBMiner, Spotfire and WinViz are three tools which are mostly used for information visualization. The brief description of these tools is given below:

a. DB-Miner: This visualization tool is developed by DBMiner Technology, Canada and is based on Microsoft Windows and also includes other data mining tools. In DBMiner, visualized data is shown as data cube and user may further analyze the data by using various data mining functions such as association, classification, characterization, prediction and clustering. This tool is based on 3-D scatter plot approach [53]. DBMiner provides a summarize view of the instances in a region in the form of single icon and the size of which is relative to the number of instances present in each region. Here, the information regarding geographical space is shown as data cube in which all individual smaller boxes further indicate regions within the given geographical space. The number of instances in each region is presented in the top left corner of the display window. The word ‘dimension’ is termed as a field in this tool. The cursor on any smaller cube can be recognized by shading and the range on the axis specified in terms of some number. The user can perform zoom in and zoom out operations on a particular region and also rotate the cube in 3-D space. DBMiner also provides the facility to look at the display from different point of view.

b. ***Spotfire:*** This tool is developed by IVEE Development AB, Goteborg, Sweden. It is based on 2-D scatter plot approach and Microsoft windows based visualization tool [53]. It visualizes the data using 2-D scatter plot and each instances of a region is presented as an icon. Spotfire provides the facility to the end user to control the screen in order to visualize the data from different perspective and performs queries on it.

c. ***WinViz:*** This tool is developed by the Information Technology Institute, Singapore. It is a Microsoft windows based visualization tool and is based on parallel co-ordinates visualization techniques [53]. WinViz is employed with additional dynamic controls and is able to effectively visualize both business and scientific data.

3.3.2. Criteria for Evaluating the Visualization Tools

Visualization Tools are evaluated on the basis of interface considerations and dataset characteristics [54]. The prime concern of interface considerations include whether the display is easy to interpret, give meaningful insight and useful for its end user. While dataset characterizes size, dimensionality, number of present clusters, the patterns of the dataset, level of background noise. Various criteria for evaluating the visualization tools are discussed below:

a. Interface Issues: Interface issues include:

Perceptually Satisfying Presentation

It is important and essential for a visualization tool to highlights the features of the dataset clearly and naturally. When the attributes are mapped into a 2-D or 3-D space, they obtain a suitable location in that space so that it can be easily understood. Thus, it is important that the display obtained by the visualization tools should be clearly visible to its end user.

Intuitiveness of the Technique

Intuitiveness of the technique refers to how easily one can interpret what is viewed. If the clusters and outliers in a dataset are identified easily without extended knowledge of the technique, then this technique is said to be intuitive. A learning curve is embedded with each visualization tool and its corresponding technique, in order to help the user for easy interpretation.

Ability to Manipulate Display Dynamically

A visualization tool is evaluated on its ability to control the display. Visualization tool is accompanied with several features such as zoom in or zoom out on a particular part of the screen and also provides control for assigning colors to particular dimension in the screen.

Ease of Use

Ease of use of visualization tool includes several factors such as flexibility of imported dataset and efficient display of the data. If there exists significant delays during display changing, the user may face difficulty of using the visualization tools. If visualization tool is not easy to use then it will not be successful in its goal of identifying patterns in the data.

b.Characteristics of the Dataset: This evaluation criterion includes:

Size of the Dataset

Different visualization techniques have different capability of displaying the datasets of varying sizes. Some handles only hundred instances and other can easily visualize large number of instances. Visualization techniques are evaluated on its capability of displaying the data of large dimension without overlap and without any loss of information.

Support for Multi-Dimensional Data

Some visualization techniques visualize many dimension of a data in a single display while others display only two, three or four dimension. Simple scatter plot visualization technique display only two or three dimension at a time.

Ability to reveal Patterns in a Dataset

The main goal of the visualization tool is to identify the patterns in a dataset. Visualization technique must be able to recognize and reveal the patterns present in the dataset. If a visualization technique is not able to reveal the patterns in the dataset then it has failed in its basic purpose.

Ability to reveal clusters

Clusters represent the relationship or association between attributes. A visualization technique must be revealed at least 2-D or 3-D clusters.

Number of Present Clusters

If more than one cluster present, then visualization technique must be able to differentiate between them in order to interpret patterns generated by them. So, it is important to identify either the clusters overlap each other or are clearly discovered as separate clusters.

Amount of Background Noise

Another important criterion for evaluating of visualization technique is to check how it deals with background noise instances. Real data contains several instances which do not produce any pattern. These instances generate background noise and create a problem in the visualization of patterns. So, it is important to evaluate the performance of visualization technique in the presence of noise.

Variance of the Clusters

It is important to observe clusters very precisely so a visualization tool must handle such variance for correct interpretation.

3.3.3. Comparison of Visualization Tools

Table 6 indicates the comparison of three visualization tools -DBMiner, WinViz and Spotfire on the basis of above mentioned criteria. The number indicates the successful level of satisfying the given criteria. For example, 1 indicates marginal successful at satisfying evaluation criteria, 2 and 3 represent successful and highly successful at satisfying criteria.

Table 6. Summary of the Comparison of the Visualization Tools

	Visualizing Tools		
Evaluation Criteria	DBMiner	Spotfire	WinViz
Core Techniques	3-D Scatter Plot	2-D Scatter Plot	Parallel Co-ordinates
Perceptually Satisfying presentation	3	3	1
Intuitiveness of Techniques	3	3	2
Ease of Use	2	3	3
Ability to Manipulate Display	2	3	3
Size of Dataset	3	2	2
Support for Multi-dimensional Data	2	2	3
Ability to reveal Cluster	2	2	3
Ability to handle Background Noise	1	3	2
Variance of Cluster	1	2	3

a) Interface Issues:

Perceptually Satisfying Presentation

DBMiner and Spotfire provide clear and natural visualization of patterns. They both highly satisfy the perceptual presentation criteria. While the patterns generated by WinViz tool are not easy to interpret and required an explanation to clarify what is represented by vertical axes and the significance of the connecting lines.

Intuitiveness of the Technique

Since DBMiner and Spotfire provide a clear presentation of the patterns so one can easily understood those patterns and easily used them further. Once the pattern generated by WinViz is understood, it becomes easy to select the instances and move axes next to each other. All the tools, DBMiner, Spotfire and WinViz support human intuition and need practice at the beginning.

Ability to Manipulate Display Dynamically

All the visualization tools provide the dynamic control of the displayed dimension to the end user. Soptfire and WinViz are comparatively superior to DBMiner and allow users to control many changes to various parameters.

Ease of Use

DBMiner is considered a complex visualization tool as compared to Spotfire and WinViz because it requires by user to generate a cube on the basis of dataset. DBMiner provides additional components along with visualization component to provide extra facilities but it further increase the complexity for the users. The database used in DBMiner must be relational database so it is complex to use. While Spotfire has the ability to accept the data in various formats and provide the dynamic control to the user so that they can easily perform zoom in and zoom out operations on particular area of the display. WinViz also provides easy access to the users. Once user understand what is represented by connecting lines and how they generate then it became a simple procedure to use.

b) Characteristics of the Data Set

Size of Data Set

DBMiner handles large number of instances as compared to Spotfire and WinViz visualization tools. It can easily represent unlimited number of instances by using a single cube which represents all the data instances in a region and one can adjust the size of cube. Whereas Spotfire and WinViz are not able to represent unlimited number of instances. Overlap or loss of information occurred due to large number of instances in Spotfire. Some feature such as Jitter in the Spotfire allows the user to deal with this problem by providing dynamic control over the display. In WinViz, the polygonal line which represented the instances can also overlap in such a way that it becomes hard for the end user to interpret the display. WinViz also has some features to overcome this limitation. It provides frequency distribution for each dimension while drawing polygonal lines so that each instance may fall in a specific value of range.

Multi-dimensional Support

DBMiner and Spotfire visualize only small dimension data. For the visualization of large dimension they are not a good choice. DBMiner visualize 3-D in a 3-D grid. The larger dimension becomes a complex task for it. Spotfire visualize 2-D in a 2-D graph. A third dimension can be visualized by coloring the icon used for each point in the 2-D graph.

Ability to reveal Patterns

DBMiner and Spotfire visualization tools are effective in recognizing outliers in a dataset. On the other hand, WinViz is not able to recognize the outliers due to overlap of lines that occur in the presence of large number of instances. DBMiner and Spotfire are able to identify clusters of three and two dimension correspondingly while WinViz is able to reveal clusters of higher dimension due to presence of parallel lines. Spotfire uses color to indicate the third dimension.

Number of Clusters Present

DBMiner, Spotfire and WinViz visualization tools are able to identify the presence of more than a single cluster even if they are in the same dimension. In DBMiner, if two clusters are present in the same dimension then both clusters may be assigned to the same cube with certain loss of information which is important to handle.

Amount of Background Noise

DBMiner is more sensitive to background noise while Spotfire and WinViz easily identify the cluster in the presence of noise.

3.4. Knowledge Integration

Traditional decision making system utilized single model or strategy to interpret the results. New decision making system combines the results produced from different models which appear as an efficient approach.

4. Research Opportunities and Challenges of KDD

The selection of a suitable Data Mining technique for a particular domain is one of the major challenges for the KDD process. It is analyzed from the literature review that no Data Mining techniques give best result for all problems [62]. Each Data Mining technique performs best in some cases but not for all domains. This problem is also known as "Superiority Problem", which means no Data Mining technique can be the best in all possible domains. The reason for this is that each technique has an explicit and implicit bias. The technique will be successful when characteristics of the problem domain satisfy its bias. If one technique performs better in one case as compared to other, it is possible that for other domain this relationship may be reversed. So, the main challenge of KDD is to decide which technique is better for a given problem. One solution of this is to combine multiple Data Mining algorithms, which is also known as ensemble approach [62].

There is a need to define the appropriate performance measure for each problem. Though there exists some common performance measurement parameters, for example, accuracy is a commonly accepted measure of performance.

Scalability of Data Mining algorithm is another major research challenge in KDD [63]. If an algorithm works well for both small and large size datasets or if there is no deterioration in its performance, then an algorithm is said to be scalable. Scalability of Data Mining algorithm is essential due to increase number of records, features or dimension, generated rules or predictive models and due to the increase demand for real time response.

Privacy and security of data that is used for the knowledge exploration is a major challenge of KDD process. So, it is essential to develop privacy and security models suitable for Data Mining approaches [63].

Mostly Data Mining algorithm uses vector-valued data. Now-a-days, the different types of data are generated by number of resources. So, there is a need to develop or extend the Data Mining techniques so that they can work efficiently with different type of dataset such as unstructured or semi structured data, time series, multimedia or hierarchical data. Another research opportunity in KDD is to develop the distributed version of Data Mining technique.

5. Conclusion

This paper provides a brief survey on pre-processing and post-processing techniques. It starts with pre-processing techniques which includes detailed description of various data cleaning approaches, imbalanced data handing and dimensionality reduction. To obtain dimensionality reduction here feature extraction and feature selection methods are mentioned. Comparative analysis and real world application of various feature selection technique is also highlighted. In this paper, various evaluation parameters such as confusion matrix, accuracy, dunn index, F-measure etc. are separately shown for classification and clustering techniques. Effective visualization is a key step of post-processing so here in this paper, three visualization tools-DBMiner, Spotfire and WinViz are discussed with their comparative analysis. Since every tool has different applicability, so their specific utility, strengths, weakness are clearly stated in this paper. This paper also highlights the research opportunities and challenges of Knowledge Discovery process.

References

[1] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "The KDD process for extracting useful knowledge from volumes of data", Communications of the ACM, vol. 39, no. 11, **(1996),** pp. 27-34**.**

[2] O. Maimon and L. Rokach, "Introduction to knowledge discovery in databases", In Data Mining and Knowledge Discovery Handbook, Springer US, **(2005)**, pp. 1-17**.**

[3] J. Han, M. Kamber and J. Pei, "Data mining: concepts and techniques", Morgan Kaufmann, **(2006).**

[4] C. Lemnaru, "Strategies for Dealing with Real World Classification Problems", Unpublished PhD thesis) Faculty of Computer Science and Automation, Universitatea Technica, Din Cluj-Napoca, **(2012)**.

[5] D. Tomar, S. Singhal, and S. Agarwal, "Weighted Least Square Twin Support Vector Machine for Imbalanced Dataset", International Journal of Database Theory and Application, vol. 7, no. 2, **(2014),** pp. 25-36**.**

[6] B. X. Wang, "Boosting support vector machine", Master Thesis, **(2005).**

[7] B. X. Wang and N. Japkowicz, "Boosting support vector machines for imbalanced data sets", Knowledge and Information Systems, vol. 25, no. 1, pp. 1-20, **(2010).**

[8] J. Laurikkala, "Instance-based data reduction for improved identification of difficult small classes", Intell Data Anal., vol. 6, no. 4, **(2002),** pp. 311–322.

[9] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study", Intelligent data analysis, vol. 6, no. 5, **(2002),** pp. 429-449.

[10] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: one-sided selection", In: Proceedings of the fourteenth international conference on machine learning, **(1997),** pp. 179–186**.**

[11] C. Ling and C. Li, "Data mining for direct marketing—specific problems and solutions", In: Proceedings of fourth international conference on knowledge discovery and data mining, **(1998),** pp. 73–79**.**

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique", arXiv preprint arXiv:1106.1813, **(2011)**.

[13] P. Domingos, "MetaCost: a general method for making classifiers cost-sensitive", In: Proceedings of the 5th ACM SIGKDD international conference on knowledge discovery and data mining, San Diego, CA, **(1999),** pp. 155–164**.**

[14] C. Elkan, "The foundations of cost-senstive learning", In: Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, WA, **(2001),** pp. 973–978**.**

[15] K. M. Ting, "An instance-weighting method to induce cost-sensitive trees", Knowledge and Data Engineering, IEEE Transactions on, vol. 14, no. 3, **(2002),** pp. 659-665**.**

[16] B. Zadrozny, J. Langford and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting", In: Proceedings of the 3rd IEEE international conference on data mining, Melbourne, FL, **(2003),** pp. 435–442**.**

[17] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem", IEEE Trans Knowl. Data Eng., vol. 18, no. 1, **(2006),** pp. 63–77**.**

[18] R. Jensen, "Combining rough and fuzzy sets for feature selection (Doctoral dissertation, University of Edinburgh)", **(2005).**

[19] L. T. Jolliffe, "Principal Component Analysis", Springer Series in Statistics, Springer-Verlag, Berlin, (**1986)**.

[20] P. Devijver and J. Kittler, "Pattern Recognition: A Statistical Approach", Prentice Hall, **(1982).**

[21] K. V. Mardia, J. T. Kent, and J. M. Bibby, "Multivariate Analysis", Probability and Mathematical Statistics Series, Academic Press, New York, **(1979).**

[22] J. H. Friedman and J. W. Tukey, “A projection pursuit algorithm for exploratory data analysis”, IEEE Transactions on Computers, vol. C-23, no. 9, **(1974)** pp. 881–890**.**

[23] J. H. Friedman and W. Stuetzle, “Projection pursuit regression”, Journal of the American Statistics Association, vol. 76, **(1981),** pp. 817–823.

[24] M. W. Richardson, “Multidimensional psychophysics. Psychological Bulletin”, vol. 35, **(1938),** pp. 659–660**.**

[25] http://en.wikipedia.org/wiki/Nonlinear_dimensionality_reduction. Accessed on 15 April 2014.

[26] J. B. Tenenbaum, V. de Silva and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction”, Science, vol. 290, no. 5500, **(2000),** pp. 2319–2323**.**

[27] S. T. Roweis and L. K. Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding”, Science, vol. 290, no. 5500, **(2000),** pp. 2323–2326**.**

[28] D. Tomar, and S. Agarwal, “Hybrid Feature Selection based Weighted Least Square Twin Support Vector Machine approach for diagnosing Breast Cancer, Hepatitis and Diabetes”, **(2014)** Unpublished Manuscript.

[29] P. Langley, “Selection of relevant features in machine learning”, In Proceedings of the AAAI Fall Symposium on Relevance, **(1994),** pp. 1–5.

[30] W. Siedlecki and J. Sklansky, “On automatic feature selection”, International Journal of Pattern Recognition and Artificial Intelligence, vol. 2, no. 2, **(1988),** pp. 197–220**.**

[31] M. Dash and H. Liu, “Feature Selection for Classification”, Intelligent Data Analysis, vol. 1, no. 3, **(1997),** pp. 131– 156**.**

[32] K. Kira and L. A. Rendell, “The feature selection problem: Traditional methods and a new algorithm”, In Proceedings of Ninth National Conference on Artificial Intelligence, **(1992),** pp. 129–134**.**

[33] H. Almuallim and T. G. Dietterich, “Learning with many irrelevant features”, In the 9th National Conference on Artificial Intelligence. MIT Press, **(1991),** pp. 547–552.

[34] H. Liu and R. Setiono, “A probabilistic approach to feature selection - a filter solution”, In Proceedings of the 9th International Conference on Industrial and Engineering Applications of AI and ES, (**1996),** pp. 284–292**.**

[35] http://en.wikipedia.org/wiki/Feature_selection. Accessed on 15th April 2014.

[36] M. A. Hall, “Correlation-based feature selection for machine learning (Doctoral dissertation, The University of Waikato)”, **(1999)**.

[37] H. H. Hsu, C. W. Hsieh, and M. D. Lu, “Hybrid feature selection by combining filters and wrappers”, Expert Systems with Applications, vol. 38, no. 7, **(2011),** pp. 8144-8150.

[38] J. Jelonek and J. Stefanowski, “Feature subset selection for classification of histological images”, Artificial Intelligence in Medicine, vol. 9, no. 3, **(1997),** pp. 227–239.

[39] S. Singh, M. Singh and M. Markou, “Feature Selection for Face Recognition based on Data Partitioning”, In Proceedings of the 15th International Conference on Pattern Recognition (ICPR 02), **(2002),** pp. 680–683.

[40] H. Handels, T. Roß, J. Kreusch, H. H. Wolff and S. P¨opple, “Feature Selection for Optimized Skin Tumor Recognition using Genetic Algorithms”, Artificial Intelligence in Medicine, vol. 16, no. 3, **(1999),** pp. 283–297**.**

[41] E. P. Xing, “Feature Selection in Microarray Analysis”, A Practical Approach to Microarray Data Analysis, Kluwer Academic Publishers, **(2003)**.

[42] M. Xiong, W. Li, J. Zhao, L. Jin and E. Boerwinkle, “Feature (Gene) Selection in Gene Expression-Based Tumor Classification”, Molecular Genetics and Metabolism, vol. 73, no. 3, **(2001),** pp. 239–247**.**

[43] W. Cede˜no and D. K. Agrafiotis, “Using particle swarms for the development of QSAR models based on k-nearest neighbor and kernel regression”, Journal of Computer-Aided Molecular Design, vol. 17, pp. 255–263, **(2003)**.

[44] Y. Saeys, S. Degroeve, D. Aeyels, P. Rouze and Y. Van De Peer, “Feature selection for splice site prediction: A new method using EDA-based feature ranking”, BMC Bioinformatics, vol. 5, doi: 10.1186/1471-2105-5-64, **(2004)**.

[45] D. Tomar and S. Agarwal, “Feature Selection based Least Square Twin Support Vector Machine for Diagnosis of Heart Disease”, International Journal of Bio-Science and Bio-Technology, vol.6, no. 2, **(2014),** pp. 69-82**.**

[46] S. Agarwal and D. Tomar, “A Feature Selection Based Model for Software Defect Prediction”, International Journal of Advanced Science and Technology, vol. 65, **(2014),** pp. 39-58.

[47] H. L. Chen, B. Yang, J. Liu and D. Y. Liu, “A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis”, Expert Systems with Applications, vol. 38, no. 7, **(2011),** pp. 9014-9022**.**

[48] Y. W. Chen and C. J. Lin, “Combining SVMs with various feature selection strategies”, Available from http://www.csie.ntu.edu.tw/~cjlin/ papers/features.pdf, **(2005).**

[49] M. F. Akay, Support vector machines combined with feature selection for breast cancer diagnosis, Expert systems with applications, vol. 36, no. 2, pp. 3240-3247, **(2009).**

[50] M. Dash, K. Choi, P. Scheuermann and H. Liu, "Feature Selection for Clustering– A Filter Solution", In Proceedings of IEEE International Conference on Data Mining (ICDM), pp. 115–122, **(2002).**

[51] G. Forman, "An Extensive Empirical Study of Feature Selection Metrics for Text Classification", Journal of Machine Learning Research, vol. 3, **(2003),** pp. 1289–1305**.**

[52] R. Jensen and Q. Shen, "Fuzzy-Rough Attribute Reduction with Application to Web Categorization", Fuzzy Sets and Systems, vol. 141, no. 3, **(2004),** pp. 469–485**.**

[53] R. Redpath, "A Comparative Study of Visualization Techniques for Data Mining (Doctoral dissertation, Monash University)", **(2000).**

[54] J. M. Chambers, W. S. Cleveland, B. Kleiner and P. A. Tukey, "Graphical Methods for Data Analysis", Chapman and Hall, New York, **(1983)**.

[55] W. S. Cleveland, "Visualizing Data; AT and T Bell Laboratories", Murray Hill, New Jersey, **(1993).**

[56] Software-DBMiner Version 4.0 Beta; DBMiner Technology Inc., British Columbia, Canada (**1998**).

[57] B. S. Everitt, "Graphical Techniques for Multivariate Data", Heinemann Educational Books Ltd. London, **(1978)**.

[58] A. Inselberg and B. Dimsdale, "Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry", Visualization '90, San Francisco, CA, pp 361-370, **(1990)**.

[59] D. A. Keim and H. Kriegel, "VisDB: Database Exploration Using Multidimensional Visualization", IEEE Computer Graphics and Applications, pp 40-49, **(1994).**

[60] H.-Y. Lee and H.-L. Ong, "Visualization Support for Data Mining", IEEE Expert, vol. 11, no. 5, **(1996),** pp. 69-75**.**

[61] D. A. Keim, "Pixel-oriented Database Visualizations", ACM Sigmod Record, Special Issue on Information Visualization, vol. 25, no. 4, **(1996)**.

[62] D. Tomar and S. A. Agarwal, "Survey on Data Mining approaches for Healthcare", International Journal of Bio-Science and Bio-Technology, vol. 5, no. 5, **(2013),** pp. 241-266**.**

[63] http://docs.rgrossman.com/tr/dmr-v8-4-5.htm. Accessed on 10th April 2014.

Authors

Divya Tomar, she is a research scholar in Information Technology Division of Indian Institute of Information Technology (IIIT), Allahabad, India under the supervision of Dr. Sonali Agarwal. Her primary research interests are Data Mining, Data Warehousing especially with the application in the area of Medical Healthcare.

Dr. Sonali Agarwal, is working as an Assistant Professor in the Information Technology Division of Indian Institute of Information Technology (IIIT), Allahabad, India. Her primary rese arch interests are in the areas of Data Mining, Data Warehousing, E Governance and Software Engineering. Her current focus in the last few years is on the research issues in Data Mining application especially in E Governance and Healthcare.