CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

The dataset may be large in size, but may not conform to the expected quality and definition. The reason for the dataset may not meet the requirement due to irrelevant and redundant data present in it. The irrelevant and redundant data constitutes a serious obstacle in identifying the pattern in the dataset. The researchers developed Feature Selection (FS) algorithm to address the problem of removal of unwanted attributes and redundant data in the dataset and prepare a required-dimensional efficient dataset. Feature subset selection is a preprocessing method of classifying and eliminating irrelevant and redundant data from dataset (Bolon-Canedo et al. 2014). Feature selection approach thus increases the performance of the classification techniques also the accuracy and capability (Wang et al. 2012). The elimination of irrelevant and redundant data from the dataset decreases the dimensionality of the dataset and brings down the time complexity significantly. In the last few decades, many feature selection algorithms have been developed. These algorithms are not designed for one particular purpose. They differ in their model as well in their function. Each of them have their own advantages and disadvantages.
The main objective of feature selection is to give good or better accuracy whilst requiring minimum features, improving learning performance, lowering computational complexity, building better generalizable models, and decreasing required storage. Feature selection improves the classification accuracy (Pregenzer et al. 1996) and reduces the attributes (Casillas et al. 2001). Feature selection algorithms can be classified into supervised, unsupervised and semi-supervised feature selection. Supervised feature selection methods can further be broadly categorized into filter models, wrapper models and embedded models. The simple method of finding the optimal subset \((d)\) from the large set \((D)\) is to evaluate all the possible item combinations \((2^D)\). However, testing all these combinations requires an exponential computational time. To address this limitation, a feature selection method has been developed. Bottom-up and top-down are the two well-known feature set generation recognized methods (Somol 1999).

3.2 ANALYSIS OF FEATURE SELECTION

Feature Subset Selection (FSS) methodology identifies the participation variables which are significant to a specific learning (Guyon and Elisseeff 2003). The Feature subset selection has two components such as search method and evaluator. The search method is the process of recognizing relevant candidate subsets. However, the evaluator will measure the behavior of a specified subset. Evaluator process is divided into two categories such as filter approach and wrapper approach.
3.2.1 Filter approach

Filter features selection approaches are using statistical methods to allocate a score to every feature based on certain criteria relevant to the problem the researcher is working on. The score is used in ranking, the features. Filter approach is one of the premature approaches for feature selection in machine learning (Mark A Hall 1998).

Figure 3.1. Feature Selection-Filter approach
Filter approaches favor to select subsets with large number of features and sometimes all the features. Consequently, a suitable threshold value is necessary to customize the filter approaches in construction of an optimal subset (Noelia and Amparo 2009). Filter approaches are computationally faster, less expensive, scalable, independent of the classifier and common than wrappers and embedded approaches (Luis Talavera 2005). Figure 3.1 illustrates filter approach. Filter approaches estimate the behavior of features or set of features by applying statistical driven approaches (Yu and Liu 2004, Javed et al. 2010). Information gain, gain ratio, chi-squared test, F-Score, ReliefF, and correlation coefficient are few filter methods.

**Information gain (IG)**

Information gain is inclined towards multivalued attributes. Information gain is a univariate technique which finds out how much information of which class is associated with the attributes (Kullback 1952). Information gain identifies the attributes with maximum information and retains them and irrelevant features or the attributes with minimum information are not included in the subset. IG is the probability that an arbitrary tuple in D belongs to class $C_i$. Let $t_i$ be the probability that a random tuple in R belongs to class $C_i$, then the entropy measure which is used as criteria in classifying a tuple in R is:

$$Info(R) = - \sum_{i=1}^{m} t_i \log_2(t_i) \quad (3.1)$$

Information needed (after using F to split R into s partitions) to classify R:

$$Info_f(R) = \sum_{i=1}^{s} \frac{|R_i|}{|R|} \times I(R_i) \quad (3.2)$$
Information gained by splitting on feature F

\[ \text{Gain}(F) = \text{Info}(R) - \text{Info}_F(R) \quad (3.3) \]

**ReliefF**

ReliefF is the best feature assessment algorithm for single label feature selection. It does not apply the comment from the classifier to allocate the weights to the features. ReliefF obtains a single set of weights that are working widely above the whole instance space. Relief does not employ the exact domain information to set feature weights (Kononenko et al. 1997). ReliefF algorithm iteratively selects an instance at random and searches for its neighbor’s in the same class which is otherwise called nearest hit and the neighbor’s from different class which otherwise called nearest hit. The difference between the current instance and its nearest hit and nearest miss is the assigned quality to every feature. ReliefF calculates the value of a feature by recurrent random sample an instance and allowing for the value of the specific feature of the nearest instance of the similar and dissimilar class. It works both on continuous and discrete class data. The unique ReliefF technique evaluates the excellence of features by how excellent their values differentiate among patterns that are near to each other. The ReliefF algorithm is specified (Kononenko 1994) as follows:

set all \( T[A] = 0 \)
for \( i = 1 \) to number of instances
begin
randomly select \( E \),
find \( N \) and \( V \)
for \( H = 1 \) to entire feature
\[ T[A] = T[A] - \text{diff}(A,E,N)/V + \text{diff}(A,E,V)/V \]

end

the algorithm represents weights \( T \), attribute \( A \), nearest value \( N \), instance \( E \), and missing value \( V \).

**Correlation-Based Feature Selection**

Correlation-Based Feature Selection (CFS) calculates the subset of attributes with the distinct analytical competence of each attribute organized with the degree of redundancy amongst them. It ensures that low inter-correlation is chosen for subsets attributes that are remarkably correlated with the class. The attributes can be a nominal attribute, numeric attribute, missing values, unfilled nominal attribute, binary attribute, unary attributes, and date attribute. The correlation between individual modules in an experiment and the external variable is identified, and the inter-correlation amongst an individual pair of modules, formerly the correlation between a combined test containing the entirety modules and the external variable can be predicted (Hall 1999).

\[ r_s = \frac{y \bar{r}_{ac}}{\sqrt{y + y(y-1) \bar{r}_{ai}}} \quad (3.4) \]

Where \( r_s \) is the \( r \) of a attributes subset \( S \) having \( y \) attributes, \( r_{ac} \) is the average attribute and class correlation, and \( r_{ai} \) is the average attribute of feature inter-correlation. Battiti (1994) CFS is an entirely instinctive algorithm. It does not necessitate the user to declare any threshold values or the number of features that need to be selected. The CFS functions on the real feature set that knowledge convinced by a machine learning algorithm by applying correlation-
based feature. Selection to select features may be made regarding the real features, but not regarding a distorted set. Significantly the CFS is a filter approach and does not experience the high computational level related to frequently appealing machine learning algorithm (Hall 1999).

3.2.2 Wrapper approach

The wrapper is an inductive algorithm. For wrapper methods, different feature subsets are selected, a predictive model is constructed for each feature subset and the feature subset which produces the model with the highest predictive performance is selected. The accuracy for different feature subsets is measured using 10-fold cross validation (Blum & Langley, 1997). Wrapper methods have typically been used for small datasets with a small number of features. It has been argued that wrapper methods are not suitable for large datasets as encountered in data mining (Hall, 1999) or datasets of high dimensionality (Yu & Liu, 2004) due to the intensive computational requirements.

Wrapper approaches denote the selection of a set of features as a search issue, where several blends are arranged, assessed and compared with various groupings. A predictive model is applied to evaluate a grouping of features and allow a score in view on model accuracy. Wrapper approaches to feature selection practice using an induction procedure to evaluate the importance of feature subsets. The basis for the wrapper is that the induction algorithms that will finally train the feature subset must deliver an improved precision than a distinct measure that requires a completely dissimilar inductive preference (Langley 1994). Wrappers approach frequently achieves improved outcomes.
than filters by reason of the circumstance that they adjust to the exact interface among an induction procedure and its training data. Though, they incline to be far slower than filters since they frequently call the induction algorithm. (Hall 1999). Figure 3.2 shows the wrapper approach, feature subset is selected by applying a learning algorithm as a portion of the estimation purpose. Wrapper approaches are computationally expensive, but simple, interacts with a classifier, and excessively slower. Wrapper approaches use search methods to insert or remove features from the dataset so that the resultant subset is a finest possible feature subsets.

![Feature Selection-Wrapper approach diagram](image)

**Figure 3.2. Feature Selection-Wrapper approach**

The wrapper approaches use a classifier as a replacement to measure the excellence of the feature subset suggested as a candidate by a search method (Kohavi and John 1997). Wrappers typically accomplish better than filter approach, but it is not suitable for large datasets as encountered in data mining
or datasets of high dimensionality due to the intensive computational requirements.

### 3.2.3 Embedded approach

Embedded approaches absorb features which include excellently contributes to the precision of the model while the model is being formed. The favorite kind of embedded feature selection approaches is regularization techniques. Regularization techniques are likewise known as penalization techniques that bring additional limitations into the optimization of a predictive procedure; for example, regression algorithm biases the model for reducing the complexity. Embedded approaches combine feature selection and model fitting into a particular optimization issue. In comparison to filter and wrapper, the learning portion and the feature selection portion is composed of embedded approaches. Decision tree learning likewise deliberated to be an embedded approach the feature selection in each iteration has been frequently through a simple filter or ranker methods by the building of the tree, and the feature selection is inserted (Ross Quinlan 1992). Adaptive boosting is also likewise considered as embedded approach, though a categorical for feature minimization may not involve throughout the training procedure (Freund and Schapire 1995).

### 3.3 ENSEMBLE CLASSIFIER

Ensemble learning merges outputs from multiple classifiers. Ensemble classifier is highly significant and useful method to increase the classification accuracy in learning algorithms (Duda et al. 2001 and Dietterich 1997). Figure 3.3 shows the ensemble classifier learning model. An ensemble based method is
called as a multiple class method which predicts by combining various classifiers. Diversity can be accomplished by applying a completely various set of classifiers and likewise in using a different training dataset with respective classifier. Ensemble-based methods or multiple classifier methods ensures standard and illustrates an actual important attention of the researchers above the previous few years (Hansen and Salmon 1990, Polikar 2006).

Multiple classifier methods are better than the single classifier due to some significant reasons. Statistics is one reason to decrease the coincidental of poor selection, feasibility of training is less due to large datasets. Resampling methods are highly effective in the small dataset. Compound decision boundary is to be learned using divide and conquer method and Data combination are valuable with heterogeneous attributes (Subrajeet Mohapatra et al. 2014).

Figure 3.3. Ensemble Classifier learning model
Bagging and boosting Breiman (1996) and Quinlan (1996) are the best prevalent ensemble learning approaches. Bagging and Boosting based ensemble classifier approaches may remove few beneficial information as they use sampling approaches to get reliable information in each of their iteration processes and they can undergo overfitting. Figure 3.3 illustrates the learning ensemble model which combines decisions of various descriptions by applying weighted voting. The importance of ensemble method describes while searching numerous independent and various decisions each of which is at less precise than random predicting, random errors revoke out, and exact decisions are strengthened.

The Bagging and Boosting based ensemble classifier approaches which frequently combine the sampling approaches with Bagging and Boosting process, they can modify the real information class distribution as they use sampling approaches to increase the smaller instances or remove, the higher class instances (Zhongbin Sun et al. 2015).

3.3.1 Bagging

Bagging combines classifiers separately trained on a bootstrap imitation of the real training set. Bagging is applied to train respectively ensemble associates by applying a randomly drawn subset of the training data.

1. for $i = 1$ to $I$  
   here $I$ is numeral of iterations 
   Bootstrap sample $S_i$ of the information 
   Learn a classifier $C_i$ from $S_i$ 
2. for each test sample
Try all classifiers \( C_i \)

Predict the class that obtains the maximum number of votes

In bagging (Breiman 1996), a training set is arbitrarily created using replacement, with sizes equivalent to the real training set. Each sample is trained by decision tree algorithms. Average of all the predictions from different trees are used which is more accurate than the prediction by a single decision tree.

### 3.3.2 Boosting

Boosting is a conventional technique for improving classifier accuracy. It combines a series of \( L \) learned classifiers, \( LC_1, LC_2, \ldots, LC_L \), to obtain an improved compound classifier, \( C^* \). In boosting, weights are allocated to each training sample, and a sequence of classifiers is learned. Once classifier \( LC_i \) is learned, the weights are updated to permit the following classifier, \( LC_{i+1} \) that takes added alertness to the misclassification errors generate by \( LC_i \). The ultimate boosted classifier, \( LC^* \) which associates the votes of the specific classifier, where the weight of individual classifier's vote is a purpose of its precision. The boosting procedure can be stretched for the estimate of continuous values Jiawei Han (2007) for every data example of the classification is achieved by identical weight voting on complete \( x \) predictors.

Voting provides important progress in the classification of accurateness and constancy. Additionally the boosting brings the ensemble of classifiers by adaptively altering the distribution of the training set based on the precision of the formerly generated classifiers and applies a classifier accurateness to weight
the selection of voting and the training samples. Adaptive boosting is one of the important boosting methods proposed by Freund and Schapire (1997). Currently, the boosting techniques and their variations have been broadly used in the computer vision and in several real-world issues. One of the very significant variations of the boosting methods is Adaptive Boosting (AdaBoost). It boosts the classification accurateness by merging various weak classification tasks into a stronger classifier. To extremely decrease the computational difficulty, the compound classifiers are merged in a cascade configuration. The compound procedure is kept merely for the additional stimulating examples (Viola and Jones 2004).

3.4 DATASET DESCRIPTIONS

Datasets are taken from various Repositories such as I2R Data Mining Department's dataset repository (Kent Ridge Bio-medical Dataset), Bioinformatic Laboratory and UCI machine learning repository. The UCI (University of California, Irvine) machine learning repository is to bring together of databases, domain models, and information producers that are applied by the machine learning group for the experiential analysis of machine learning procedures. The documentation was formed as an ftp archive in 1987 by David Aha and fellow scholar at UC Irvine (Arthur Asuncion and David Newman 2007). Table 3.1 describes the dataset information. Multiple datasets are the datasets with mixtures of various datasets like medical datasets, forecasting datasets, chemical datasets and written numerals (‘0’-’9’) extracted from a group of Dutch utility maps datasets. The dataset consists of maximum numbers of attributes and instances. The attribute characteristics are also categorized into nominal and numeric datasets.
Lung Cancer Dataset:

This dataset is extracted from the UCI machine learning repository. The dataset consist information of 3 types of pathological lung cancers. Information on the individual variables is not available. Attribute 1 is the class label and 59 predictive attributes are nominal, taking on integer values 0-3. The value for 5th and 39th attribute are found be -1 and 4 which are outside the range 0-3 for which no explanation is found.

Prostate_tumorVSNormal dataset:

This dataset is taken from the bioinformatics laboratory. This classification model is constructed from gene expression profiles of 10 biopsies.
of androgen-independent primary prostate tumor and 10 biopsies of primary untreated androgen-dependent tumors.

**Ozone Level Detection Dataset:**

The Air Quality Index (AQI) indicates how the air is polluted. AQI reports about ground-level ozone, particle pollution, carbon dioxide, sulfur dioxide and nitrogen dioxide. People if exposed to air of AQI score above 150 can be affected by respiratory or cardiovascular problems. Ozone level dataset available in the UCI machine learning repository contains ground level ozone data. The data in the Ozone level dataset is grouped as eight-hour peak set (eighthr.data), and one hour peak set (onehr.data). The attributes in the dataset that begins with T represent the temperature and with WS denotes wind speed. This dataset contains ozone level data collected over the period 1998 to 2004 at the Brazoria, Houston, and Galveston region.

**Ovarian Dataset:**

The dataset is taken from I2R Data Mining Department's dataset repository (Kent Ridge Bio-medical Dataset), was published in (Emanuel F Petricoin III et al. 2002). The raw spectral data of each instance, have the comparative amplitude of the intensity at each molecular mass/charge (M/Z) character. There are complete 15155 M/Z features. The formula normalized the intensity standards: \( NR = (R - \text{Min})/(\text{Max} - \text{Min}) \), where NV is the normalized value, R the raw value, Max the maximum intensity and Min the minimum intensity. The normalization is carried over entirely the 253 instances for all 15155 M/Z features. The intensity value is to suit in the interior range of 0 to 1 after normalization.
SRBCT Dataset:

The small round blue cell tumors are four different childhood tumors appearing very similar in appearance on routine histology. The similarity between the childhood tumor cells makes correct clinical diagnosis extremely challenging. SRBCTs (small round blue cell tumors) dataset is taken from the Bioinformatic Laboratory. The source of data is National Human Genome Research Institute (Khan et al.). SRBCT is a classification model designed to identify the four childhood tumors neuroblastoma (NB), Ewing's family of tumors (EWS), non-Hodgkin lymphoma (in the case Burkitt's lymphoma, BL) and rhabdomyosarcoma (RMS). Out of the 88 samples in the dataset 63 is used for training and 25 samples for testing. The platform used in the dataset is cDNA microarrays.

Gas Dataset:

Alexander Vergara constructed gas dataset. Alexander Vergara collected 13910 measurements gathered from 16 chemical sensors fixed in a gas delivery platform facility situated at the ChemoSignals Laboratory in the BioCircuits Institute, University of California, USA. The dataset represents data collected over the period January 2007 to February 2011. The dataset comprises concentrations of six distinct pure gaseous substances, namely Ammonia, Acetaldehyde, Acetone, Ethylene, Ethanol, and Toluene.
LSVT-voice-rehabilitation:

Athanasios Tsanas of the University of Oxford used 309 speech signal processing algorithms and built LSVT (Lee Silverman Voice Treatment) with 126 samples from 14 participants. LSVT was developed for treating the voice and speech disorders in individuals with Parkinson disease and other neurological disorders. LVST compare vowel phonations of PD subjects with the vowel phonations of speech expert and decide whether PD subject voices as acceptable or unacceptable. 90 percent of PD patients treated by LSVT showed good improvement.

MFeat Fourier dataset:

Multiple feature Digit dataset consist features of handwritten numerals (0-9). Robert P.W Duin, Delft University of Technology, Netherlands digitized 200 patterns of each numerals as binary images. mfeat- Fourier, mfeat-profile correlations, mfeat- karhunen, mfeat- pixel, mfeat-zernike moments, mfeat-morphological features are the feature sets used to represent the numerals.

3.5 RESEARCH DESIGN

This section describes the research definition including the details of the specific focus area of the research, and the various procedures used for the research and analysis. Methodology also summarizes the scope, test data and ethics adopted for this research work.
This research work progresses on three main phases that are described below and shows in figure 3.4. Phase I deals with hybrid feature subset selection for multiple dataset using decision tree based classifiers. Phase II deals with effective AdaBoost (eAdaBoost) algorithm for decision tree classifier. Phase III deals with the mutual model, the feature subsets selected in Phase-I are incorporated along with boosting algorithms in phase II.

Phase I discusses a hybrid feature subset selection technique with the decision tree based classification algorithm. Feature selection is one of the most significant procedures in machine learning algorithms. It is particularly used to improve the performance and prediction accuracy for complex data classification. Hybrid feature subset selection is superlative data preprocessing techniques used to remove the irrelevant data. The feature selected using Information Gain (IG) is combined with the features selected from ReliefF which generates the resultant feature subset. Then the resultant feature subset is in turn combined with a Correlation-based Feature Selection (CFS) method to generate the aggregated feature subset. Intersection and exclusive OR (XOR) are two fusion methods used to combine the feature subsets. The classification accuracies and the statistical test comparisons are made with various feature selection methods namely CFS, IG, ReliefF, Intersection and XOR. To perform classification accuracy on the aggregated feature subset, different Decision trees based classification algorithm such as C4.5, Decision Stumps, Naive Bayes Tree, and Random Forest with ten-fold cross-validation is deployed. To check the prediction accuracy of the proposed work eight different multiple UCI (University of California, Irvine) machine learning datasets has been used with minimum to maximum numbers of features. The main objective of the hybrid feature selection is to improve the classification accuracy, prediction and to
reduce the execution time using standard datasets. Phase I is elaborately explained in the chapter 4.

Phase II introduces a novel ensemble and meta classifier method named eAdaBoost (Effective Adaptive Boosting) which is developed by enhancing the existing AdaBoost algorithm and to handle the time complexity and also to produce the best classification accuracy. The ensemble Meta classifier eAdaBoost algorithm is used to improve the classification accuracy. The eAdaBoost reduces the error rate when compared with the existing methods and generates the best accuracy by reweighing each feature for further process. The comparison results of an extensive experimental evaluation of the effective AdaBoosting method are explained using the same UCI machine learning repository datasets.

The accuracy of the classifiers and statistical test comparisons are made with various boosting algorithms such as AdaBoost, Real AdaBoost, MultiBoostAB and proposed method eAdaBoost. The suggested eAdaBoost has been also implemented with different decision tree classifiers like C4.5, Decision Stump, NB Tree and Random Forest. The algorithm has been computed with various dataset, with different weight thresholds and the performance is analyzed. The proposed method produces better results using random forest and NB tree as base classifier than the decision stump and C4.5 classifiers for few datasets. The eAdaBoost gives better classification accuracy, prediction accuracy, and the execution time is also less when compared with other classifiers. This phase of the works is explained in detailed in chapter 5.
Phase III deals with the mutual model, the feature subsets selected in Phase I are incorporated along with boosting algorithms in phase II. Therefore, the aggregated feature subsets from phase I are functioned with two boosting algorithms namely AdaBoost and proposed method eAdaBoost to find the classification accuracy. To find enhanced classification accuracies, various Decision tree based classification algorithms is used such as C4.5, Decision Stumps, Naïve Bayes Tree and Random forest with 10 fold cross validations. Therefore, the same UCI machine learning datasets are used for the experimentation process. The output of a phase III shows the better classification accuracy and very less time consumption for all selected decision tree classifiers. Phase III is explained in detail in chapter 6.
Figure 3.4. Framework of Hybrid Feature Subset Selection using effective AdaBoost Algorithm
3.6 CLASSIFIER EVALUATION METRICS

Classification methods performance is evaluated using the classification evaluation metrics: Accuracy, Error Rate, Sensitivity, Specificity, Precision and Recall, and F-measures. Figure 3.5 illustrates the evaluation of classification. The classification or prediction accuracy is derived from feature subset selection approach and boosting algorithms with various classifiers such as C4.5, DecisionStumps, Random forest and NB tree classifiers.

\[
\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}
\]  

(3.5)

**Figure 3.5. Evaluation of classification**

The classifier accuracy is that proportion of test set tuples that are appropriately classified or identified (TP+TN) to the total number of instances in the test set (TP+TN+FP+FN)
Therefore the accuracy of predicted values is figured out in the figure as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The evaluation of classification system is a hypothesis is used to conclude classification of instances in the test set; concluded classification is compared to known classification.

3.6.1 Evaluating Classifier Accuracy

The most straightforward type of evaluation will be as far as classification accuracy. In other words it is the number of instances whose class the classifier can correctly predict. The classifier accuracy can be evaluated by holdout technique, repeated holdout technique, cross-validation technique, Leave-one-out cross-validation (LOOCV), bootstrap methods, and Parameter tuning.

3.6.1.1 Evaluation criteria

Classification or predictive precision denotes the capability of the model in predicting the class label of fresh or formerly unseen data. The evaluation criteria can be analyzed in the various aspects like:

- Accuracy: percentage of testing set instances suitably classified by the classifier
- Speed: states the computational costs associated with creating and applying the model
- Robustness: deals with the capability of the model to make the right expectations specified noisy data or by missing values
- Scalability: discusses the capability to build the model capably given significant amount of data
- Interpretability: states the level of comprehend and perception that is delivered by the model
- Simplicity: this refers to the size of the decision tree and rule compactness

### 3.6.1.2 Cross-Validation Approaches

In k-fold cross-validation, the original dataset is first partitioned into k subsets of equal size, $S_1$……$S_k$. In first iteration sets $S_2$……$S_k$ form the training set and $S_1$, is the test set. Consequently in the second iteration sets $S_1$ and $S_3$……$S_k$ form the training set, and $S_2$ is the test set; and so on. Ultimately, on the $k^{th}$ fold, $S_1$……$S_{k-1}$ form the training set and $S_k$ is the test set. The results all the ‘folds’ are averaged to arrive the accuracy. Usually the original dataset is partitioned into 10 equal subsets of equal sizes so that any overlapping of test sets is avoided and the existing data is efficiently used.