Implementation of FAST using Map-Reduce

Mrs. Sandhya Waghere, Dipali Morankar, Kajal Sawant, Tejal Shah

Information Technology, Pune University
PCCOE, Pune-411044,India

Abstract

The process of finding interesting patterns and knowledge from huge amount of data is known as Data Mining. Association rule mining is One of the most important techniques in this field is association rule mining. Association rule mining is a process which is meant to discover frequent patterns, correlations, associations, or causal structures from datasets. Feature selection is the process of selecting a subset of relevant features from large amount of data. Feature selection involves identifying a subset of the most useful features that reduces the size of original dataset. A feature selection algorithm may be evaluated from both the efficiency and effectiveness points of view. Efficiency is based on the time required to find a subset of features and the effectiveness is related to the quality of the subset of features. Based on these criteria, a fast clustering-based feature selection algorithm, FAST, is used[18]. The FAST algorithm works in two steps. In the first step, features are divided into clusters. In the second step, the most representative feature that is strongly related to target classes is selected from each cluster to form a subset of features. Features in different clusters are relatively independent; the clustering-based strategy of FAST has a high probability of producing a subset of useful and independent features. To ensure the efficiency of FAST, we adopt the efficient minimum-spanning tree clustering method. The efficiency and effectiveness of the FAST algorithm are evaluated through an empirical study.

Keywords– *Data Mining, FAST, Redundant Features, Irrelevant features, Clustering, Representative Features.*

1. INTRODUCTION

Data mining is the extraction of hidden predictive results from large databases. Feature selection is defined as process of selecting subset of relevant feature. The method for using a feature selection technique is that data may contain many redundant or irrelevant features. With the aim of choosing a subset of good features with respect to the target concepts, feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy and improving result comprehensibility [2][3]. A feature selection algorithm may be evaluated from both the efficiency and effectiveness points of view. While the efficiency concerns the time required to find a subset of features, the effectiveness is related to the quality of the subset of features[3]. In machine learning and statistics, feature selection also known as variable selection, attribute selection or variable subset selection. It is the process of selecting a subset of relevant features for use in model construction. The central assumption when using a feature selection technique is that the data contains many redundant or irrelevant features. Redundant features are those which provide no more information than the currently selected features, and irrelevant features provide no useful information in any context. Feature selection techniques are a subset of the more general field of feature extraction. Feature extraction creates new features from functions of the original features, whereas feature selection returns a subset of the features. Feature selection techniques are often used in domains where there are many features and comparatively few samples or data points. Feature selection is used to cluster the related data in database. Feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, improves the efficiency. It improves learning accuracy and performance of classifiers. The aim of feature selection is to determine a feature subset as small as possible. It is the essential preprocessing step prior to applying data mining tasks. It selects the subset of original features, without any loss of useful information. It removes irrelevant and redundant features for reducing data dimensionality. As a result it improves the mining accuracy, reduces the computation time and enhances result comprehensibility[2][3]. On applying mining tasks to the reduced feature subset produces, the same result as with original high- dimensional dataset.

1.1 Procedure of Feature Selection

In the process of feature selection, irrelevant and redundant features or noise in the data may be hinder in many situations, because they are not relevant and important with respect to the class concept such as microarray data analysis. When the

number of samples is much less than the features, then machine learning gets particularly difficult, because the search space will be sparsely populated. Therefore, the model will not able to differentiate accurately between noise and relevant data[10]. The general procedure for feature selection has four key steps as shown in Figure.

- Subset Generation
- Evaluation of Subset
- Stopping Criteria
- Result Validation

1.1.1 Subset Generation

Subset generation is a heuristic search in which each state specifies a candidate subset for evaluation in the search space. Two basic issues determine the nature of the subset generation process. First, successor generation decides the search starting point, which influences the search direction. From the set of full features, first we must determine the starting point in feature space, which in turn influences the direction of search. The search for feature subsets can start with no features. Or, it can start with all (full) features. To decide the search starting points at each state, forward, backward, compound, weighting, and random methods may be considered.

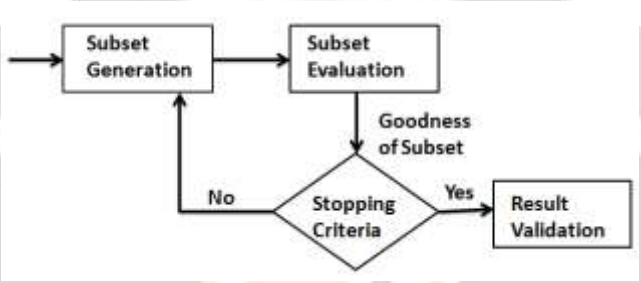


Fig. 1.1 Procedure of Feature Subset Selection[10]

Second, search organization is responsible for the feature selection process with a specific strategy, such as sequential search, exponential search or random search. Theoretically, the best subset of features can be found by evaluating all the possible subsets, which is known as exhaustive search. But an exhaustive search of the feature space needs to search all of n 2 possible subsets of n features, so it is almost always impractical when we meet large number of features. Therefore, we have to consider a more realistic and practical approach. Several search procedures that are easier to implement have been developed, but they are not guaranteed to find the optimal subset of features. These search procedures differ in their computational cost and the optimality of the subsets they find.

1.1.2. Evaluation of Subset

After generating subsets of candidate features, we need to evaluate them. A newly generated subset must be evaluated by a certain evaluation criteria. Therefore, many evaluation criteria are used to determine the goodness of the candidate subset of the features. Base on their dependency on mining algorithms, evaluation criteria can be categorized into groups: independent and dependent criteria. Independent criteria exploit the essential characteristics of the training data without involving any mining algorithms to evaluate the goodness of a feature set or feature. And dependent criteria involve predetermined mining algorithms for feature selection to select features based on the performance of the mining algorithm applied to the selected subset of features.

1.1.3. Stopping Criteria

Finally, we must decide the criteria for halting (stopping) the search. For example, we can stop adding or removing features when none of the alternatives improves the estimate of classification accuracy, or we can stop when the number of selected features reaches a pre-determined threshold. We can then choose the best subset among the candidates we have encountered during the search.

1.1.4. Result Validation

Feature selection process stops at validation procedure. It is not the part of feature selection process, but feature selection method must be validate by carrying out different tests and comparisons with previously established results or comparison with the results of competing methods using artificial datasets, real world datasets, or both.

2. LITERATURE REVIEW

[1] M. Dash, H. Liu, and H. Motoda, "Consistency Based Feature Selection," Proc. Fourth Pacific Asia Conf. Knowledge Discovery and Data Mining, pp. 98-109, 2000.

They stated that the feature selection algorithms uses various measures to determine the usefulness and effectiveness of search result. The paper mainly concentrated on consistency measure for feature selection. They have explained the consistency, its properties and comparison with other present measures for feature selection. Also we studied how to calculate inconsistency measure, distance measure, consistency measure, information measure for better search.

[2] Lei Yu, Huan Liu," Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution", 2003.

Feature Selection for high dimensional data is challenging task. This paper provides a study of feature redundancy in high-dimensional data and proposes a novel correlation based approach to feature selection within the filter model. Classical linear correlation helps to remove features with near zero linear correlation to the class and reduce redundancy among selected features. It uses FCBF algorithm. This algorithm firstly calculates symmetrical uncertainty SU and then calculate list of symmetrical and relevant data.

In this paper a new feature selection algorithm FCBF is implemented and evaluated through extensive experiments comparing with related feature selection algorithms. The feature selection results are further verified by applying two different classification algorithms to data with and without feature selection. Our approach demonstrates its efficiency and effectiveness in dealing with high dimensional data for classification.

[3] Qinbao Song, Jingjie Ni and Guangtao Wang, "A Fast Clustering-Based Feature Subset Selection Algorithm for High Dimensional Data", 2013.

Irrelevant features, along with redundant features, severely affect the accuracy of the learning machines Thus, feature subset selection should be able to identify and remove as much of the irrelevant and redundant information as possible. Moreover, "good feature subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other."

FAST algorithm efficiently and effectively deal with both irrelevant and redundant features, and obtain a good feature subset. In FAST algorithm, it involves (i) the construction of the minimum spanning tree (MST) from a weighted complete graph; (ii) the partitioning of the MST into a forest with each tree representing a cluster and (iii) the selection of representative features from the clusters .Each cluster is treated as a single feature and thus dimensionality is drastically reduced.

[4] Nitin B Chopade, Beena S Khade, "A New Clustering Based Algorithm for Feature Subset Selection". 2014, 5272-5275.

This paper contains the main idea of the FAST clustering based feature selection algorithm and its step for working of algorithm. They have also explained the main challenge that if more than one feature are joint and they suit the target feature then it can be treated as relevant. Feature interface is the new challenge for identifying the applicable feature. FAST algorithm extract only targeted features out of many features. They don't measure the irrelevant and redundant data because irrelevant and redundant data affects the competence and effectiveness of the algorithm. It also explains distributed clustering and time complexity of prim's algorithm.

[5] R.P.L.DURGABAI "Feature Selection using ReliefF Algorithm",2014.

This paper states the ReliefF algorithm, extension of Relief algorithm. The original relief can only deal with nominal and numerical attributes. However, it cannot deal with incomplete data and it is limited to two-class problems. ReliefF algorithm overcomes these problems. This algorithm is not limited to two class problems, is more robust and can deal with incomplete and noisy data. ReliefF randomly selects an instance Ri , but then searches for k of its nearest neighbors from the same class, called nearest hits Hj , and also k nearest neighbors from each of the different classes, called nearest misses Mj(C). It updates the quality estimation W[A] for all attributes A depending on their values for Ri, hits Hj and misses Mj(C). The update formula is similar to that of Relief, except that they average the contribution of all the hits and all the misses. The contribution for each class of the misses is weighted with the prior probability of that class P(C) (estimated from the training set). Selection of k hits and misses is the basic difference to Relief and ensures greater robustness of the algorithm concerning noise. User defined parameter k controls the locality of the estimates. For most purposes it can be safely set to 10. To deal with incomplete data we change the diff function. Missing values of attributes are treated probabilistically.

[6] R.Munieswari,"A Survey on Feature Selection Using FAST Approach to Reduce High Dimensional Data",IJETT, Volume 8 Number 5- Feb 2014.

It stated the survey on different feature selection techniques or algorithms including wrapper, filter, fast clustering algorithm, hybrid approach and relief algorithm. The comparison among all these algorithms is explained. The Relief algorithm concerns

with giving weight to each cluster. Then reorder the cluster according to max weight. When the cluster weight crossed the threshold value, the particular cluster is taken as feature set. All other methods are studied in previous and other papers mentioned above. They have also stated the FAST algorithm with use of MST and gives advantage of FAST over all other techniques.

[7] S. Francisca Rosario, Dr. K. Thangadurai, "RELIEF: Feature Selection Approach", 2015.
This paper states Relief algorithm, which was first described by Kira and Rendell [KIRA92] as a simple, fast, and effective approach to attribute weighing. The Relief algorithm weights each feature according to its relevance to the class. Initially, all weights are set to zero and then updated iteratively. In each iteration, this non-deterministic algorithm chooses a random instance i in the dataset and estimates how well each feature value of this instance distinguishes between instances close to i. In this process two groups of instances are selected: some closest instances belonging to the same class and some belonging to a different class. With these instances, Relief will iteratively update the weight of each feature and it differentiates data points from different classes while, simultaneously, recognizing data points from the same class. At the end, a certain number of features with the highest weights is selected. In an alternative version, a threshold may be used in such a way that only the features with weights above this value are selected.
In RELIEF algorithm an instance based attribute ranking scheme. It deals with incomplete, noisy and multiclass datasets. It behaves unpredictably for very small expected counts, which are common in text classification. In text classification word features occurred rarely.

[8] P. Ramasita, 2 S. Rama Sree, "An Approach for Hybrid Cluster Based Feature Selection on High Dimensional Data", 2015.
There are two types for feature selection methods: Filter method, the classifiers are used in which it separates the of features and in the Wrapper method, the features are chosen by the classifier. The filter model is used due to its computational effectiveness and also improves the capability. The wrapper methods calculate the variables which have, dissimilar like filter model, to identify the achievable interaction between the variables. The two main disadvantages of these models are 1.The increases' ended appropriate threat when the number of explanation is insufficient, 2.The significant calculation time when the number of variables is big. A hybrid model has newly projected to deal with high dimensional data. In this model, first it compute the features subsets depends on their given precedence and then cross justification is taken for ultimate best subset diagonally different precedence. These algorithms mainly focus to combine filter and wrapper model to reach best possible datasets with the minimum relevance and maximum redundancy and they can be obtained by using Entropy and Gain.

[9] Surendar Singh, Ashutosh Kumar Singh, "Web Spam Feature Subset Selection using CFS-PSO":2017.
In this paper, they stated CFS algorithm which is a type of filter algorithm that choose features according correlation based function. The preference of this function is to select subgroups that contain features that are extraordinarily related with the class but uncorrelated with each other. Unessential features ought to be disregarded on the grounds that they will have low relationship with the class while repetitive features are screened out as they will be exceptionally related with at least one of the rest of features. The acknowledgment of a feature will rely upon the degree to which it predicts classes in territories of the instance space not as of now anticipated by different features.

3. PROPOSED METHODOLOGY

The feature subset selection should be able to identify and remove as much of the irrelevant and redundant information as possible. We develop a Feature Selection algorithm which can efficiently and effectively deal with both irrelevant and redundant features, and obtain a good feature subset. We achieve this through a new feature selection framework which composed of the two connected components of irrelevant feature removal and redundant feature elimination. The former obtains features relevant to the target concept by eliminating irrelevant ones, and the latter removes redundant features from relevant ones via choosing representatives from different feature clusters, and thus produces the final subset. The irrelevant feature removal is straightforward once the right relevance measure is defined or selected, while the redundant feature elimination is a bit of sophisticated. In our proposed FAST algorithm[18], it involves 1) the construction of the minimum spanning tree from a weighted complete graph; 2) the partitioning of the MST into a forest with each tree representing a cluster; and 3) the selection of representative features from the clusters.

Advantages:

- Subsets of good feature contain features which are highly correlated with target class, and others are uncorrelated with each other.

- The efficiency and effectiveness both are deal with irrelevant and redundant features, and obtain a good feature subset.
- Less training set and less memory will occupy by handling semi supervised process.
- Alike data can't be miss in cluster data by using pairwise constraint.
- Overlapping avoided by using maximum margin cluster process.

3.1 Hybrid Systems

The main goal of hybrid systems for feature selection is to extract the good characteristics of filters and wrappers and combine them in one single solution. Hybrid algorithms achieve this behavior usually by pre-evaluating the features with a filter in a way to reduce the search space to be considered by the subsequent wrapper. The term "hybrid" refers to the fact that two different evaluation methods are used, a filter-type of evaluation and the classifier accuracy evaluation methods.

3.2 Flowchart of FAST algorithm

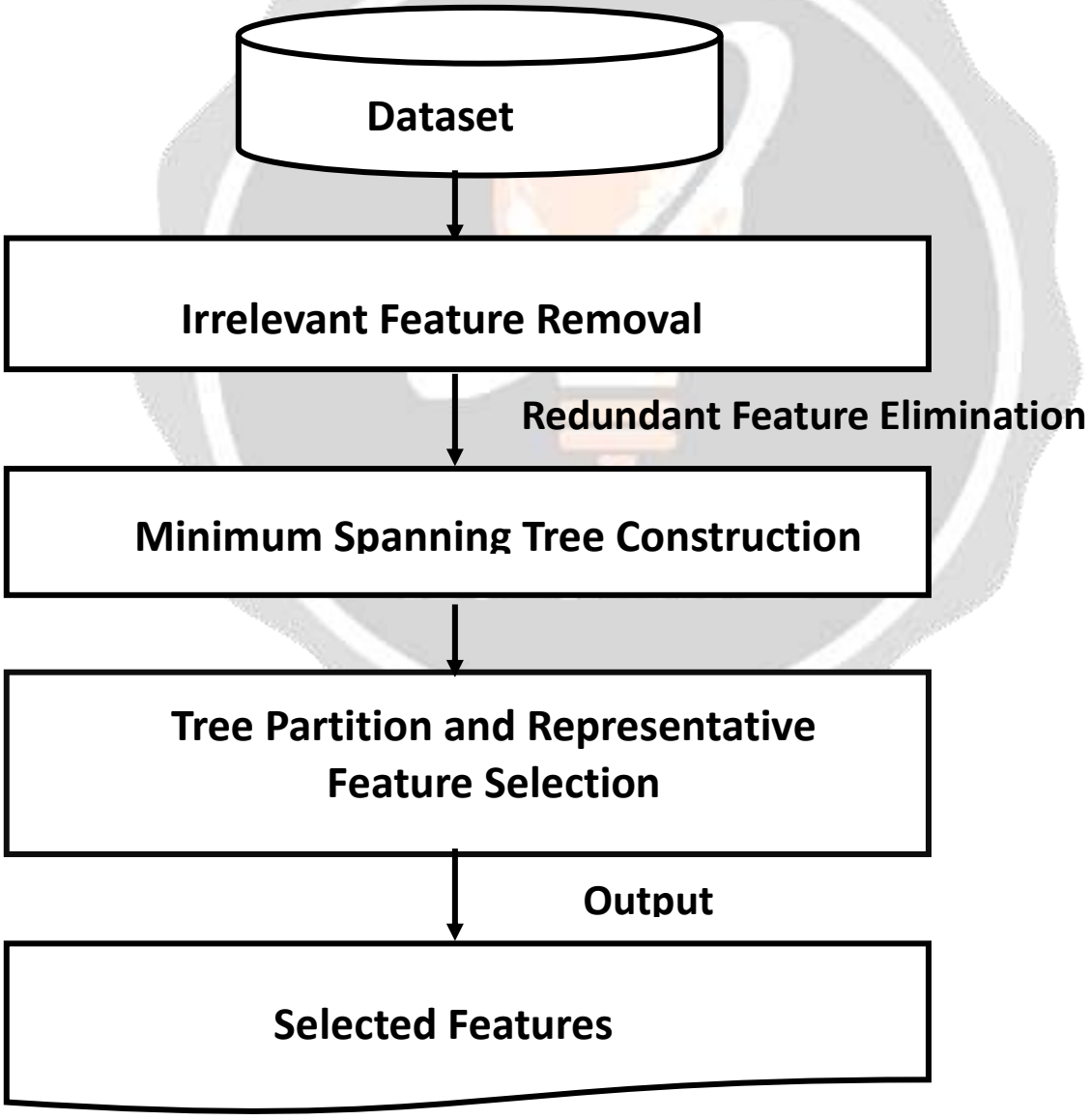


Fig.3.2.1 Flowchart of FAST algorithm

3.3 System Architecture

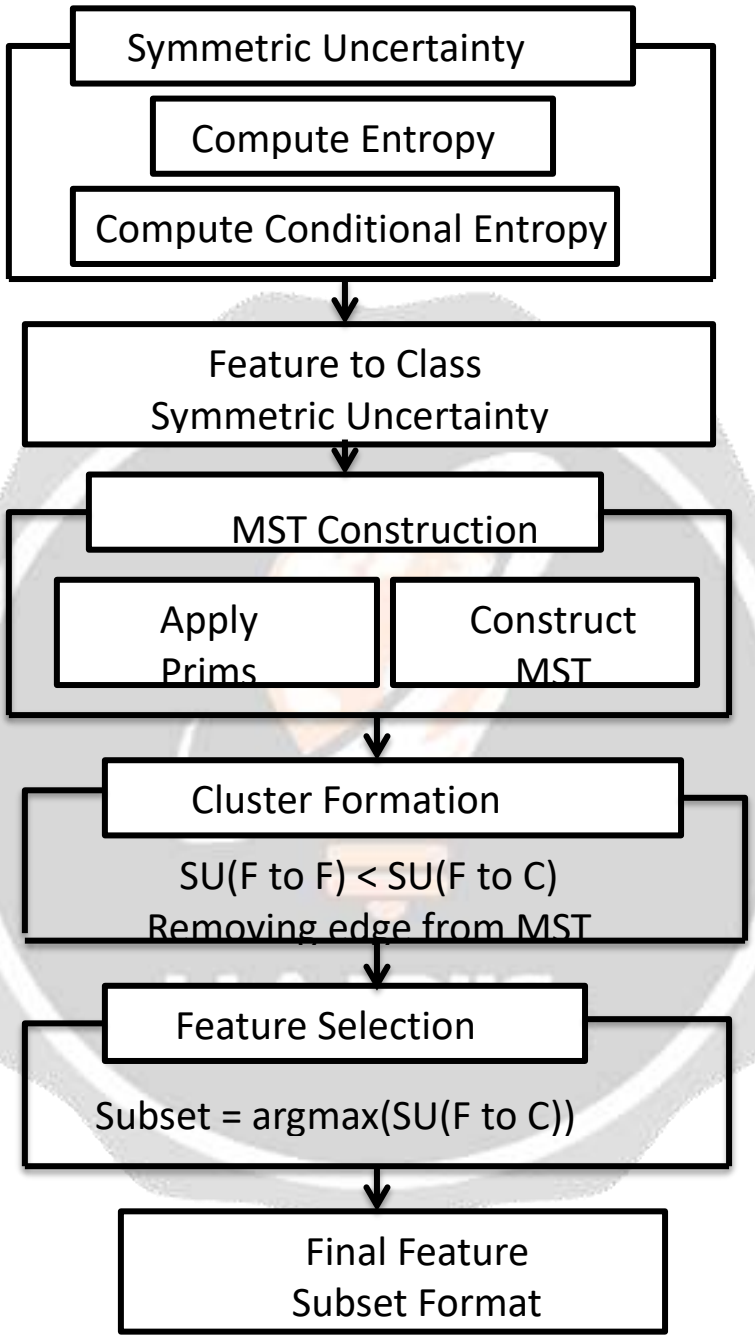


Fig.3.3.1 System Architecture

3.4 Framework

i. Construction of MST:

The Minimum Spanning Tree (MST) is constructed by using Prim's algorithm. This algorithm works better than existing Kruskal's algorithm The symmetric uncertainty (SU) is derived from the mutual information by normalizing it to the entropies of feature values or feature values and target classes, and has been used to evaluate the goodness of

features for classification Therefore, choose symmetric uncertainty as the measure of correlation between either two features or a feature and the target concept.

ii. Partitioning MST:

After MST remove the edges whose weights are smaller than both of the TRelevance from MST deletion results in two disconnected trees T1 and T2. The set of vertices in any one of the final trees to be V (T) have the property that for each pair of vertices guarantees the features in V (T) are redundant. Removing all the unnecessary edges, a forest is obtained. Each tree T Forest represents a cluster that is denoted as V (T), which is the vertex set of T as well. As the features in each cluster are redundant, so for each cluster (T) choose a representative feature.

3.5 Clustering High-Dimensional Data:

Most clustering methods are designed for clustering low-dimensional data and encounter challenges when the dimensionality of the data grows really high. This is because when the dimensionality increases, usually only a small number of dimensions are relevant to certain clusters, but data in the irrelevant dimensions may produce much noise and mask the real clusters to be discovered[19]. Moreover, when dimensionality increases, data usually become increasingly sparse because the data points are likely located in different dimensional subspaces. To overcome this difficulty, we may consider using feature (or attribute) transformation and feature (or attribute) selection techniques. Feature transformation methods, such as principal component analysis and singular value decomposition, transform the data onto a smaller space while generally preserving the original relative distance between objects. They summarize data by creating linear combinations of the attributes, and may discover hidden structures in the data. However, such techniques do not actually remove any of the original attributes from analysis. This is problematic when there are a large number of irrelevant attributes. The irrelevant information may mask the real clusters, even after transformation. Moreover, the transformed features (attributes) are often difficult to interpret, making the clustering results less useful. Thus, feature transformation is only suited to data sets where most of the dimensions are relevant to the clustering task. Unfortunately, real-world data sets tend to have many highly correlated, or redundant, dimensions. Another way of tackling the curse of dimensionality is to try to remove some of the dimensions. Attribute subset selection (or feature subset selection) is commonly used for data reduction by removing irrelevant or redundant dimensions (or attributes). Given a set of attributes, attribute subset selection finds the subset ofattributes that are most relevant to the data mining task. Attribute subset selection involves searching through various attribute subsets and evaluating these subsets using certain criteria. It is most commonly performed by supervised learning—the most relevant set of attributes are found with respect to the given class labels. It can also be performed by an unsupervised process, such as entropy analysis, which is based on the property that entropy tends to be low for data that contain tight clusters. Other evaluation functions, such as category utility, may also be used. Subspace clustering is an extension to attribute subset selection that has shown its strength at highdimensional clustering. It is based on the observation that different subspaces may contain different, meaningful clusters. Subspace clustering searches for groups of clusters within different subspaces of the same data set. The problem becomes how to find such subspace clusters effectively and efficiently.

Entropy: A decision tree is built top-down from a root node and involves dividing the data into subsets that contain instances with identical values (homogenous). To determine the homogeneity of a sample algorithm ID3 is used. Entropy is zero if the sample is completely homogeneous and the entropy is one if sample is an equally divided.

We need to determine two types of entropy using frequency tables to build a decision tree as follows:

a) Entropy using the frequency table of one attributes:

$$E(S) = \sum - pi \log2\ pi \quad i=1$$

b) Entropy using the frequency table of two attributes:

$$E(T, X) = \sum P(c)E(c)\ C \in X$$

Symmetric Uncertainty:

For measuring correlation between either two features or a feature and the target concept we choose symmetric uncertainty.

$$SU(X, Y) = 2 * Gain(X \mid Y) / H(X) + H(Y)$$

Where,

1. H (X) = Entropy of a discrete random variable X. Suppose p(x) is the prior probabilities for all values of X, H (X) is defined by

$$H(X) = - \sum p(x) \log2\ p(x)\ x£X$$

2. Gain (X|Y) = amount by which the entropy of Y decreases. It reflects the additional information about Y provided by X and is called information gain.

$$\text{Gain}(X|Y) = H(X) - H(X|Y)$$
$$= H(Y) - H(Y|X)$$

Where H (X|Y) is the conditional entropy which quantifies the remaining entropy of a random variable X given that the value of another random variable Y is known.

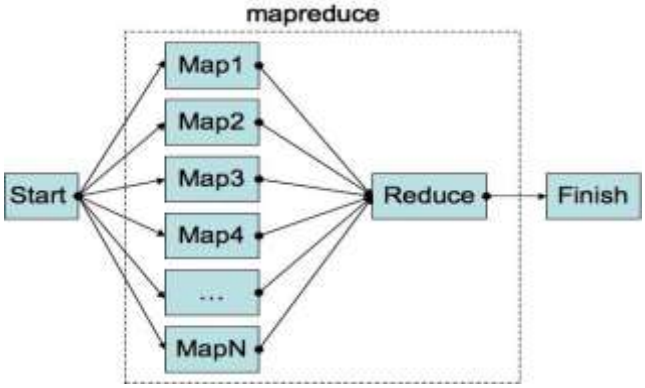
Suppose p(x) is the prior probabilities for all value of X and p(x|y) is the posterior probabilities of X given the values of Y, H(X|Y) is defined by

$$H(X|Y) = -\sum_{y \in Y} p(y) \sum_{x \in X} p(x|y) \log_2 p(x|y)$$

Information gain is a symmetrical measure. That is the amount of information gained about X after observing Y is equal to the amount of information gain about Y after observing X. This ensures that the order of two variables will not affect the value of the measure.

3.6 MapReduce Paradigm

MapReduce[2] is a programming paradigm designed to analyse large volumes of data in a parallel fashion. Its goal is to process data in a scalable way, and to seamlessly adapt to the available computational resources. A MapReduce job transforms lists of input data elements into lists of output data elements. This process happens twice in a program, once for the Map step and once for the Reduce step. Those two steps are executed sequentially, and the Reduce step begins once the Map step is completed. In the Map step, the data elements are provided as a list of key-value objects. Each element of that list is loaded, one at a time, into a function called mapper. The mapper transforms the input, and outputs any number of intermediate key-value objects. The original data is not modified, and the mapper output is a list of new objects. In the Reduce step, intermediate objects that share the same key are grouped together by a shuffling process, and form the input to a function called reducer. The reducer is invoked as many times as there are keys, and its value is an iterator over the related grouped intermediate values. Mappers and reducers run on some or all of the nodes in the cluster in an isolated environment, i.e. each function is not aware of the other ones and their task is equivalent in every node. Each mapper loads the set of files local to that machine and processes it. This design choice allows the framework to scale without any constraints about the number of nodes in the cluster. An overview of the MapReduce paradigm is reported in figure 1. Algorithms written in MapReduce scale with the cluster size, and Execution Time (ET) can be decreased by increasing the number of nodes. The design of the algorithm and the data layout are important factors impacting ET. This is the case of the equi-join relational operator in MapReduce, where two of the most commonly used join strategies are Repartition Join and Broadcast Join. The former performs the join operation on the reducers,therefore requiring all data to be moved acrossthe network. The latter broadcasts the smaller table across all nodes, and the operation is run as map-only job. It therefore avoids the transmission of the larger table across the network[23] .In ET terms, jobs perform better in MapReduce when transformations are executed locally during the Map step, and when the amount of information transferred during the shuffling step is minimized. In particular, MapReduce is very well-suited for associative and commutative operators, such as sum and multiplication. These can indeed be partially processed using an intermediate Combine step, which can be applied between the Map and Reduce stages. The combiner is an optional functionality in MapReduce. It locally aggregates mapper output objects before they are sent over the network. It operates by taking as input. The intermediate key-value objects produced by the mappers, and output key-value pairs for the Reduce step. This optional process allows to reduce data transfer over the network, therefore reducing the global ET of the job.



Hadoop: Using the MapReduce algorithm Hadoop runs the applications, where the data is processed in parallel on different CPU nodes[18]. In short, Hadoop framework is capable enough to develop applications capable of running on clusters of computers and they could perform complete statistical analysis for huge amounts of data. A Hadoop frame-worked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage. The term MapReduce actually refers to the following two different tasks that Hadoop perform.

The Map Task: This is the beginning task, which takes input data and converts it into a set of data, where distinctive elements are broken down into tuples (key/value pairs)

The Reduce Task: This task takes the output from a map task as input and combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map task.

4. CONCLUSION

In this paper, we have studied an Efficient FAST clustering-based feature subset selection algorithm for high dimensional data which improves the efficiency of the time required to find a subset of features. The algorithm involves 1) removing irrelevant features, 2) constructing a minimum spanning tree from relative ones, and 3) partitioning the MST and selecting representative features. In the FAST algorithm, a cluster consists of features. Each cluster is treated as a single feature and thus dimensionality is drastically reduced and classification accuracy is improved by improving performance of classification.

5. REFERENCES

[1] M. Dash, H. Liu, and H. Motoda, "Consistency Based Feature Selection," Proc. Fourth Pacific Asia Conf. Knowledge Discovery and Data Mining, pp. 98-109, 2000.

[2] Lei Yu, Huan Liu," Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution", 2003.

[3] Qinbao Song, Jingjie Ni and Guangtao Wang, "A Fast Clustering-Based Feature Subset Selection Algorithm for High Dimensional Data", 2013.

[4] Nitin B Chopade, Beena S Khade, "A New Clustering Based Algorithm for Feature Subset Selection". 2014, 5272-5275.

[5] R.P.L.DURGABAI "Feature Selection using ReliefF Algorithm", 2014.

[6] R.Munieswari,"A Survey on Feature Selection Using FAST Approach to Reduce High Dimensional Data",IJETT, Volume 8 Number 5- Feb 2014.

[7] S. Francisca Rosario, Dr. K. Thangadurai, "RELIEF: Feature Selection Approach", 2015.

[8] P. Ramasita, 2 S. Rama Sree, "An Approach for Hybrid Cluster Based Feature Selection on High Dimensional Data", 2015.

[9] Surendar Singh, Ashutosh Kumar Singh, "Web Spam Feature Subset Selection using CFS-PSO":2017**.**

[10] Vipin Kumar and Sonajharia Minz , "Feature Selection: A literature Review",2015.

[11] Verónica Bolón-Canedo ,Noelia Sánchez-Maroño, Amparo Alonso-Betanzos., "Feature selection for high-dimensional data".

[12] S. Saranya, R. Munieswari, "A Survey on Feature Selection Using FAST Approach to Reduce High Dimensional Data"

[13] Mr. Akshay S. Agrawal , "Clustering Based Feature Subset Selection Algorithm Using FAST"

[14] Vishnu Tore1, Prof. P.M.Chawan, "FAST Clustering Based Feature Subset Selection Algorithm for High Dimensional Data",2016

[15] Prof. M. MUSTHAFA, R.ROKIT KUMAR, "Feature Subset Selection for High Dimensional Data using FAST and improve energy efficiency" 2014.

[16] Pawan Gupta, Susheel Jain, Anurag Jain, "A Review Of Fast Clustering-Based Feature Subset Selection Algorithm", 2014

[17] Vishnu M. Tore, Prof. P. M. Chawan, Prof. S. A. Khedkar, Prof. Kishor Dongarwar, "Survey On: Comparison of Clustering Based Feature Subset Selection Algorithms for High Dimensional Data"2016.

[18] Sandhya S Waghere, Pothuraju Rajarajeswari, "Parallel Frequent Dataset Mining and Feature Subset Selection for High Dimensional Data on Hadoop using Map-Reduce" *in International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 12, Number 18, pp. 7783-7789,2017.

[19] Sandhya S. Waghere, PothurajuRajarajeswari, "A Survey on Achieving Best Knowledge from Frequent Item set Mining using Fidoop" *in International Journal of Computer Applications* (0975 – 8887) Volume 171 – No. 9, August 2017.