Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-3, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

A Feature Subset Selection Algorithm for High Dimensional Data Based on Fast Clustering

Prof. Ganesh Shelke, Akanksha Sonawane, Nidhi Bongale,
Sheetal Dhanwe & Shivani Bhosale
Assistant Professor, VIIT, Pune
Student, VIIT, Pune

***Abstract**: Feature selection consists of recognizing a subset of the most valuable features which produces compatible results same as the original entire set of features. A feature selection algorithm can be evaluated from both the efficiency and effectiveness. The efficiency is evaluated by the time required to find a subset of features and the effectiveness is related to the quality of the subset of features. A fast clustering-based feature selection algorithm(FAST) is proposed based on efficiency and effectiveness. The FAST algorithm involves two steps. The first step involves dividing features into clusters by using methods of graph-theoretic clustering. The second step involves the most representative feature that is strongly related to target classes is selected from each cluster to form a subset of features. There are relatively independent features in different clusters, the clustering-based strategy of FAST has a high probability of producing a subset of useful and independent features. We adopt the efficient minimum spanning tree(MST) clustering method to ensure the efficiency of FAST. For this we use MST(Minimum Spanning Tree using Kruskal's algorithm clustering based method.*

Introduction

The goal is to choose the subset of useful features with respect to the target concepts, feature subset selection is an effective way for reducing dimensionality, remove unnecessary data, increasing learning accuracy, and improve result comprehensibility. The feature subset selection methods are proposed for machine learning applications. They can be divided into four parts: 1)The Embedded 2) Wrapper 3) Filter and 4)Hybrid approaches. The embedded methods involves feature selection as the part of training process and they are usually depend on given learning algorithms, and therefore may be more efficient than the other three types Traditional machine learning algorithms working like decision trees or artificial neural networks are examples of embedded approaches. The wrapper methods useful the predictive accuracy of a predetermined learning algorithm to determine the wellness of the selected subsets, the accuracy of the learning algorithms is mostly high. But,the generality of the selected features is limited and the computational complex is large.

Related Work

An Feature subset selection is the process of identifying and removing all possible irrelevant and redundant features because of irrelevant features do not contribute to the predictive accuracy and redundant features do not redound to getting a better predictor for that they provide already present information in other features. Of the feature subset selection algorithms, some can effectively remove features which are irrelevant but fail to handle redundant features yet some of others can remove the irrelevant while taking care of the redundant features. FAST algorithm is our proposed algorithm which falls into the second group. Classically, feature subset selection research has focused on searching for relevant features. A famous example is Relief, that weighs each feature according to its ability of discriminating instances under different targets based on function of distance-based criteria. Relief is ineffective at eliminating redundant features as two predictive although highly correlated features are likely both to be highly weighted. Relief-F extends Relief, enabling this method to noisy work and data sets which are incomplete and to deal with multi-class problems, yet still cannot identify redundant features. CFS is done by the hypothesis that a good feature subset is one which contains features highly correlated with the target yet correlated which each other. FCBF is a fast filter method that can identify relevant features and redundancy among relevant features without correlation analysis. CMIM iteratively picks features that maximize their mutual information with the class to predict, to the response of any feature already picked conditionally. Different from these algorithms , our proposed the FAST algorithm using the clustering-based method to choose features. Recently, hierarchical clustering which is adopted in word selection in the context of text classification. Distributional clustering is used to cluster words into groups based either on their participation in particular grammatical relations with other words by Pereira et al or on the distribution of class labels in association with each word by Baker and McCallum. As distributional clustering of words are agglomerative in nature, and result in high computational cost and suboptimal word clusters, Dhillon et al. proposed a new information-theoretic divisive algorithm for word clustering which is applied to text classification. Butterworth et al. proposed to features of using a special metric of Barthelemy-Montjardet distance, and then using the dendrogram of the resulting cluster hierarchy to choose the many of the relevant attributes. Unfortunately, the measure of cluster evaluation based on Barthelemy-Montjardet distance does not identify a feature subset which allows the classifiers to improve their performance accuracy originally. Further, feature selection methods, the obtained accuracy is lower. Hierarchical clustering also is used to select features on spectral data.

Van Dijck and Van Hulle proposed a hybrid filter or wrapper feature subset selection algorithm for regression. Krier et al. presented a method for combining hierarchical constrained clustering of spectral variables and selection of clusters by using mutual information. Feature clustering method is similar to that of Van Dijck and Van Hulle except that the former constraits every cluster to contain consecutive features only. Both methods employed agglomerative hierarchical clustering to eliminate redundant features. Different from these hierarchical clustering-based algorithms, FAST algorithm uses minimum spanning tree-based method for clustering features. Simultaneously, it does not assume that data points are grouped around centers or separated by a regular geometric curve. Mainly, our proposed FAST does not limit to some specific types of data.

3. Feature Subset Selection Algorithm

3.1 Framework and Definitions

Irrelevant features and redundant features affect the machine learning accuracy severely. Hence, selection of feature selection should be able to identify and remove as much of the irrelevant and redundant information as possible. “Good feature subsets contain features which are highly correlated with the class, yet uncorrelated with each other.”

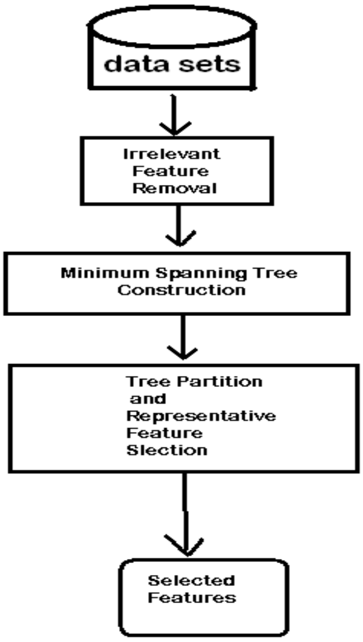
data sets
Irrelevant Feature Removal
Minimum Spanning Tree Construction
Tree Partition and Representative Feature Slection
Selected Features

Fig. : Flowchart for Implementing Fast Algorithm

Keeping these in mind, we develop a novel algorithm which can efficiently and effectively manage 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 couple of connected components of irrelevant feature removal and eliminating redundant features. Relevance is obtained to the target concept by eliminating irrelevant ones, and the removal of redundant features from relevant ones by choosing representatives from different feature clusters, and hence produces the final subset.

The irrelevant feature removal is straightforward once the correct relevance measure is selected, while the redundant feature elimination is little sophisticated. In our proposed FAST algorithm, it includes constructing minimum spanning tree from a weighted complete graph, the partitioning of the MST into a forest where each tree representing a cluster; and the selection of representative features from the clusters.

In order to more precisely introduce the algorithm, and as our proposed feature subset selection framework includes irrelevant feature removal and redundant feature elimination, the traditional definitions of relevant and redundant features are presented firstly , then provide our definitions based on variable correlation as follows.

Suppose F to be the full set of features, F_i belongs to F be a feature, Si= F - $\{F_i\}$. S_i’ is a subset of S_i. Let s_i’ bea value-assignment of all features in S_i’, f_i value-assignment of feature Fi, and c a value-assignment of the target concept C.

Relevant feature (definition) : F_i is relevant to the target concept C if and only if there exists some s_i, f_i, and c, such that, for probability of S_i'=s_i' , F_i=f_i is greater than zero.

P(C=c| S_i'=s_i', F_i=f_i) is not equal to P(C=c| S_i'=s_i')

Otherwise, feature Fi is an irrelevant feature. Definition of relevance indicates that there are two kinds of relevant features due to different Si. First, when Si ‘=Si, as Fi is directly relevant to the target concept, Secondly when Si ‘ $\underline{C}$ Si. It tells that Fi is irrelevant to the target concept. But, the definition shows that feature Fi is relevant when using which describes the target concept. The reason behind is that either Fi is interactive with Si or redundant with Si -Si. Hence, we say Fi is indirectly relevant to the target concept.
Redundant features do not contribute to getting better interpreting ability to the target concept as most of the information contained in redundant features is already present in other.

Definition (Redundent Feature): Let S be - a set of features, a feature in S is redundant if and only if it has a Markov Blanket within S.
Relevant features have very strong correlation with target concept and hence are necessary for a best subset, while redundant features are not necessary because their values are completely correlated with one another. Hence, notions of feature redundancy and feature relevance are in terms of feature correlation and feature-target concept correlation.
Mutual information measures quantity of the distribution of the feature values and target classes which differ from statistical independence. This is a nonlinear estimation of correlation between feature values and target classes or feature values. The symmetric uncertainty (SU) is derived from the information by normalizing it to the entropies of feature values or feature values and target classes, and has been used to evaluate the quality of features for classification by a number of researchers. Hence, we select symmetric uncertainty as the measure of correlation between either couple of features or a feature and the target concept.

feature and the target concept.

The symmetric uncertainty is defined as follows:

SU(X,Y)= 2* Gain(X|Y) / (H(X)+H(Y))

Where,

1. H(X) is the entropy of a discrete random variable X.
2. Gain(X|Y)is the amount by which the entropy of Y decreases. It reflects the additional information about Y provided by X and is called the informationgain which is given by

Gain(X|Y)= H(X)-H(X|Y)
Gain(Y|X)= H(Y)-H(Y|X)

Where H(X|Y) is the conditional entropy that quantifies the remaining entropy of a random variable X given that the value of another random variable Y is known.
Pair of variables is treated symmetrically by symmetric uncertainty , it compensates for information gain's bias toward variables with more values and normalizes its value to the range 0 to 1. Knowledge of the value of either one completely predicts the value of the other is indicated by value 1 and the value 0 reveals that X and Y are independent. Although the measure which is based on entropy handles nominal or discrete variables, they can also deal with continuous features, if the values are in advance discretized properly.

Given SU(X,Y) the symmetric uncertainty of variables X and Y , the T-Relevance between a feature and the target concept C, the correlation F-Correlation between a pair of features, the feature redundance F-Redundancy and the representative feature R-Feature of a feature cluster can be defined as follows.

T-Relevance : The relevance between the feature Fi and the target concept C is referred to as the T-Relevance of Fi and C which is denoted by SU(F_i,C).
F-Correlation : The correlation between any pair of features Fi and Fj
R-Feature: A feature Fi is a representative feature of the cluster S (i.e., Fi is a R-Feature) if and only if,
Fi = argmax (SU(Fi,C))

This means the feature with the strongest T-Relevance can act as a R-Feature for all the features in the cluster. According to the definitions mentioned above, feature subset selection can be the process that identifies and retains the strong T-Relevance features and selects R-Features from feature clusters.

3.2 Algorithm:

```
Inputs : Given data set D($F_1$,$F_2$, ………, $F_M$, C)
T : the T-Relevance threshold
Output: S- Selected feature subset

//---Part 1: Irrelevant Feature Removal---//
For i=1 to m do
    T-Relevance =SU($F_i$,C)
    If T-Relevance > T then
        S=S U {$F_i$};

//---Part 2: Minimum spanning tree construction---//
G=NULL;
For each pair of features {$F_i$',$F_j$'} C S do
    F-Correlation=SU($F_i$',$F_j$')
```

```
  Add $F_i'$ and/or $F_j'$ to G with F-Correlation as weight of corresponding edge;

minSpannTree=Kruskal(G);
  //using kruskal's algorithm for generation of minimum spanning tree


//---Part 3: Tree Partition and representative Feature Selection---//
Forest = minSpannTree
For each edge $E_{i,j}$ belongs to Forest do
  If SU( $F_i'$, $F_j'$)< SU( $F_i'$, C) ^ SU($F_i'$,$f_j'$)< SU($F_j'$,c) then
              Forest = Forest- $E_{i,j}$

S=NULL
For each tree Ti belongs to Forest do
  $F^j_R$ = argmaxSU($F_k'$,C)
  S=S U { $F^j_R$ };
Return S
```

Time Complexity:The first part of the algorithm has a linear time complexity O(m) in terms of the number of features m. In the first part, assuming k(1 to m)features are selected as relevant ones. Only one feature is selected when k=1. Hence, there is no need to continue the remaining parts of the algorithm, and the complexity is O(m). When $1<k\leq m$, the second part of the algorithm first constructs a complete graph from relevant features and the complexity is $O(k^2)$, and then an MST from the graph is generated using Kruskal's algorithm whose time complexity is $O(k^2)$. The third part partitions the MST and selects the representative features with the complexity of O(k). Therefore, when $1<k\leq m$, the complexity of the algorithm is$O(m+k^2)$. This means when $k\leq m^{1/2}$, FAST has linear complexity O(m), while obtains the worst complexity $O(m^2)$ when k=m. However, k is heuristically set to be $[m^{1/2}*\lg m]$ in the implementation of FAST. So the complexity is $O(m*\lg^2 m)$.

4. EMPIRICAL STUDY:

4.1 Data Source

For the purposes of rank the performance and effective of our proposed FAST algorithm, verifying whether it was not the method is potentially used in practice, and allowed other researchers to confirm our results, 35 publicly possibale data sets1 were use. The numbers of features of the 35 data sets alter from 37 to 49,52 with a mean of 7,874. The dimensionality of the 54.3 percent data files top 5,000, of which 28.6 percent data files have more than 10,000 subsets. The 35 data files cover a range of application scope such as text, image and bio microarray data classified.Table 12 views the corresponding statistical info. Note that for the data texts or the files with continuous-value features, the well-known off-the-shelf MDL method are used to discretize the continuous values.Summary of the 35 Benchmark Data files.

4.2 Emperial Setup:

To evaluate the performance of our proposed FAST algorithm and divide it with other feature selection algorithms in a fair and modest way, we set up our experimental study as follows:

1) The proposed algorithm is divide with five different types of representative feature selection algorithms They are :

1) FCBF 2) ReliefF 3) CFS 4) Consist and last one is 5) FOCUS-SF respectively.

FCBF and ReliefF rank or survey features individually.For FCBF, in the experiments, we set the relevance threshold to be the SU value of the bm=log mcth ranking the feature for each data set (m is the number of features in a given data set) as suggested by Yu and Liu.ReliefF founds for closest neighbors of items of different classes and weights features according to how well they differentiate details of
different classes.

The other three feature selection algorithms are based upon the subset evaluation. CFS exploits best first search based upon the evaluation of a subset that contains features highly connect with the target concept, yet error with each other. The Consist method found for the minimal subset that can not the connect to the each other classes as consistently as the full colletion can under best first search strategy. FOCUS-SF is a variation of FOCUS. FOCUS has the same evaluation strategy as Consist, but it examines all subsets of features. Considering the time efficiency, FOUCS-SF replaces exhaustive in found FOCUS with one by one forward selection. For our proposed FAST algorithm, we heuristically set to be the SU value of the contain feature for each data set.

2) Four different types of classification algorithms can be:employed to classify data texts before and after feature selection They are

1) the probability-based Naive Bayes 2) the tree-based 3) the instance-based lazy learning algorithm and last on is 4) the rule-based RIPPER respectively..

Naive Bayes utilizes a probabilistic method for classified by multiplying the solel probabilities of every feature-value pair. This algorithm assumes that the independence among the features and even then provides good classification results. Decision tree learning algorithm C4.5 is an extension of ID3 that accounts for unavailable values, continuous attribute value ranges, thin of decision trees, rule derivation.

The tree comprises of nodes (features) that are selected by information..

Instance-based learner IB1 is a single-nearest algorithm classifed entities taking the class of the nearest associated vectors in the training set via distance metrics. It is the simple among the algorithms used in our algorithem study. Inductive rule learner Repeated Incremental reduced to Produce Error Reduction is a rule learner that defines a rule-baded detection model and seeks to improve it.

3. When evaluating the performance of the feature subset selection algorithms have four metrics

1) The proportion of selected features 2) the time to obtain the feature subset 3) the classification accuracy and last is 4) the Win/Draw/Loss record are used.

The proportion of selected features is the of the scale number of features selected by a feature selection algorithm to the original number of features of a data set or the files. The Win/Draw/Loss record presents three values on a given count i.e. the numbers of data files for which our proposed algorithm FAST obtains good , equal, and poor performance than other five feature selection algorithms, respectively. The count can be the proportion subsets of selected features, the runtime to obtain a features subset the classification variety respectively.

4.3 Empirical Procedure work:

In order to make the good use of the collection of the data and obtain stable results, a ðM ¼ 5Þ ðN ¼ 10Þ-cross-validation strategy is use. That is for each data files , each feature subset selectionalgorithm and each classification algorithm, the 10-fold cross-validation is repeated M = 5 times along with every time the order of the instances of the data files being randomly.

This is because many of the algorithms display order effects in that certain way have been ordered dramatically improve or decrease the performance Randomizing the order of the inputs that canhelp diminish the order effects.

In the experiment for every feature subset selection algorithm, we obtain MN feature subsets and the corresponding runtime Time with each data files. Average and Time, we can obtain the number of selected features further the proportion of selected features and the corresponding runtime for each other feature selection algorithm oneach data files.

For each classification algorithm, we obtain MN classification exact time for each feature selection algorithm and each data files. Average these Accuracy, we canobtain mean accuracy of each classified algorithm under each feature selection algorithm and each data set files. Theprocedure Experimental Process view the details.

5. Conclusion

For high dimensional data an well defined FAST clustering-based feature subset selection algorithm improves the efficiency of the time required to find a subset of features. The algorithm includes 1) removing irrelevant features, 2)constructing a minimum spanning tree(MST) from relative ones, and 3) partitioning the MST and selecting representative features. In the defined algorithm, a cluster consists of features. Each cluster is treated as single feature and thus dimensionality is drastically reduced and improved the classification accuracy.

.

6. References

[1] Qinbao Song, Jingjie Ni and Guangtao Wang, “A Fast ClusteringBased Feature Subset Selection Algorithm for High Dimensional Data”, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING VOL:25 NO:1 YEAR 2013.

[2] H. Almuallim and T.G. Dietterich, “Algorithms for IdentifyingRelevant Features,” Proc. Ninth Canadian Conf. Artificial Intelligence, pp. 38-45, 1992.

[3]M. Dash, H. Liu, Feature selection methods for classification, Intelligent Data Analysis: An Internat. J. 1 (3) (1997).

[4] Forman G., “An extensive empirical study of feature selection metrics for text classification”, Journal of Machine Learning Research, 3, pp 1289- 1305,2003.

[5] Huan Liu, “Toward Integrating Feature Selection Algorithms for Classification and Clustering”, IEEE transactions on knowledge and data engineering, VOL. 17, NO. 4, April 2005.

[6] Krier C, Francois D, Rossi F and Verleysen M, “Feature clustering and mutual information for the selection of variables in spectral data”, In Proc European Symposium on Artificial Neural Networks Advances in Computational Intelligence and Learning, pp 157-162,2007.

[7] Lei Yu, Huan Liu, “Efficient Feature Selection via Analysis of Relevance and Redundancy”, Journal of Machine Learning Research, 5, 1205–1224, 2004.

[8] Lei Yu, Huan Liu,” Efficiently Handling Feature Redundancy in High Dimensional Data”, ACM, August 27, 2003.

[9] Lei Yu, Huan Liu, “Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution”, Proceedings of the Twentieth International Conference on Machine Learning, 2003.

[10] Lei Yu, Huan Liu,” Redundancy Based Feature Selection for Microarray Data”, ACM, August 2004.

[11] J. Biesiada and W. Duch, “Features Election for High-Dimensional data a Pearson Redundancy Based Filter,” Advances in Soft Computing, vol. 45, pp. 242-249, 2008.

[12] Guyon I. and Elisseeff A., An introduction to variable and feature selection,Journal of Machine Learning Research, 3, pp 1157-1182, 2003.

[13] L.D. Baker and A.K. McCallum, “Distributional Clustering of Words for Text Classification,” Proc. 21st Ann. Int’l ACM SIGIR Conf. Research and Development in information Retrieval, pp. 96- 103, 1998.

[14] Jihoon Yang and Vasant Honavar, ”Feature Subset Selection Using A Genetic Algorithm Artificial Intelligence Research Group,

[15] H. Liu and R. Setiono, “A Probabilistic Approach to Feature Selection: A Filter Solution,” Proc. 13th Int’l Conf. Machine Learning, pp. 319-327, 1996.