Asian Journal of Computer Science and Technology (AJCST)
Vol.2.No.2 2014 pp 31-35.
available at: www.goniv.com
Paper Received :05-03-2014
Paper Published:28-03-2014
Paper Reviewed by: 1. John Arhter 2. Hendry Goyal
Editor : Prof. P.Muthukumar

FEATURE SUBSET SELECTION USING GRAPH-THEORETIC CLUSTERING METHOD

P.Umadevi,L.Priya

ABSTRACT

Feature subset selection can be viewed as the process of identifying and removing many irrelevant and redundant features. Even though some can eliminate irrelevant features but fails to handle redundant features. Clustering-based feature subset selection algorithm for high dimensional data involves removing irrelevant features, constructing a minimum spanning tree from relative ones, partitioning the MST and selecting representative features. A feature selection algorithm may be evaluated from both the efficiency and effectiveness. The FAST algorithm works in two steps, in the first step features are divided into clusters by using graph-theoretic clustering method. 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. To evaluate the “information content” of each individual feature with regard to the output. Redundancy reduction may be used in unsupervised methods of data analysis.

Index Terms—Feature subset selection, filter method, feature clustering, graph-based clustering

1. INTRODUCTION

Clustering high-dimensional data is the cluster analysis of data with anywhere from a few dozen to many thousands of dimensions. Such high-dimensional data spaces are often encountered in areas such as medicine, where DNA microarray technology can produce a large number of measurements at once, and the clustering of text documents, where, if a word-frequency vector is used, the number of dimensions equals the size of the dictionary. Multiple dimensions are hard to think in, impossible to visualize, and, due to the exponential growth of the number of possible values with each dimension, complete enumeration of all subspaces becomes intractable with increasing dimensionality.

This problem is known as the curse of dimensionality. The concept of distance becomes less precise as the number of dimensions grows, since the distance between any two points in a given dataset converges. The discrimination of the nearest and farthest point in particular becomes meaningless. The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. It can be shown that there is no absolute “best” criterion which would be independent of the final aim of the clustering. Consequently, it is the user which must supply this criterion, in such a way that the result of the clustering will suit their needs. Clustering real-world data sets is often hampered by the so-called curse of dimensionality: many real-world data sets consist of a very high dimensional feature space. In general, most of the common algorithms fail to generate meaningful results because of the inherent sparsity of the data space. Usually, clusters cannot be found in the original feature space because several features may be irrelevant for clustering due to correlation and/or redundancy.

However, clusters are usually embedded in lower dimensional subspaces. In addition, different sets of features may be relevant for different sets of objects. Thus, objects can often be clustered differently in varying subspaces of the original feature space. A cluster is intended to group objects that are related, based on observations of their attribute's values. However, given a large number of attributes some of the attributes will usually not be meaningful for a given cluster. For example, in newborn screening a cluster of samples might identify newborns that share similar blood values, which might lead to insights about the relevance of certain blood values for a disease. But for different diseases, different blood values might form a cluster, and other values might be uncorrelated. This is known as the local feature relevance problem: different clusters might be found in different subspaces, so a global filtering of attributes is not sufficient.

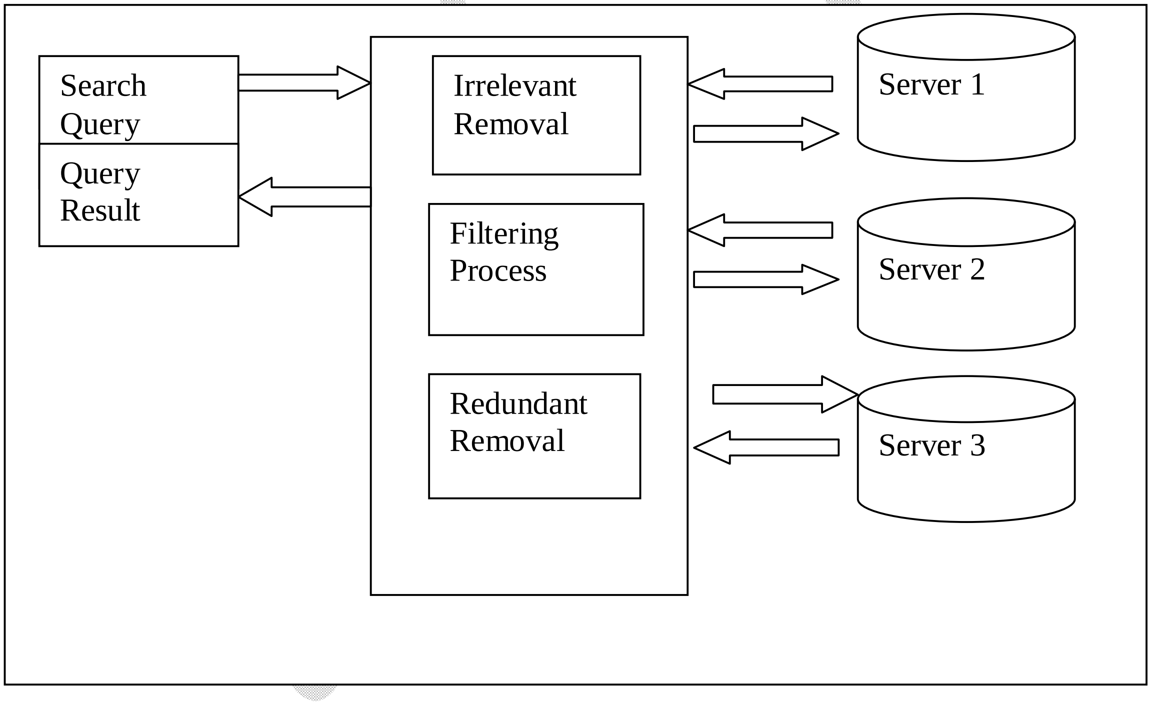
Given a large number of attributes, it is likely that some attributes are correlated. Hence, clusters might exist in arbitrarily oriented subspaces.

2. RELATED WORK

Feature subset selection can be viewed as the process of identifying and removing as many irrelevant and redundant features as possible. This is because irrelevant features do not contribute to the predictive accuracy and redundant features do not redound to getting a better predictor for that they provide mostly information which is already present in other features. Of the many feature subset selection algorithms, some can effectively eliminate irrelevant features but fail to handle redundant features yet some of others can eliminate the irrelevant while taking care of the redundant features.

according to its ability to discriminate instances under different targets based on distance-based criteria function. However, Relief is ineffective at removing redundant features as two predictive but highly weighted. Relief-F extends Relief, enabling this method to work with noisy and incomplete data sets and to deal with multiclass problems, but still cannot identity redundant features.

Feature selection involves identifying a subset of the most useful features that produces compatible results as the original entire set of features. 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. 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.



3. PROPOSED WORK

Proposed system falls into the second group. Traditionally, feature subset selection research has focused on searching for relevant features. A well-known example is Relief which weighs each feature

Upload File or Text

The class can process multiple files uploaded with the file or text form. The files that were uploaded successfully are moved to a given directory. The class may reject files that exceed a given size limit. The descriptions are picked from the value of a form text field that is submitted with the file field data. The details of file name, description and size is stored in a separate file with a given file name, as a serialized array of data that can be retrieved to provide the necessary information to generate pages on which the uploaded files are displayed.

Irrelevant Feature Removal

The former obtains features relevant to the target concept by eliminating irrelevant from different feature clusters. The irrelevant feature removalis straightforward once the right relevance measure is defined or selected. The relevant features have strong correlation with target concept so are always necessary for a best subset. Feature subset selection can be the process that identifies and retains the strong irrelevant features and selects relevant from feature clusters. This is a nonlinear estimation of correlation between feature values or feature values and target classes.

Graph-Based Clustering (minimum spanning tree)

The general graph-theoretic clustering is simple: Compute a neighborhood graph of instances, then delete any edge in the graph that is much longer/shorter (according to some criterion) than its neighbors. The result is a forest and each tree in the forest represents a cluster. Then apply graph theoretic clustering methods to features. The features are divided into clusters by using graph-theoretic clustering methods. The construction of the minimum spanning tree (MST) from a weighted complete graph.

Redundant Feature Removal

The latter removes redundant features from relevant ones via choosing representatives from different feature clusters, and thus produces the final subset. The redundant feature eliminationis a bit of sophisticated.Redundant features are assembled in a cluster and a representative feature can be taken out of the cluster. Thus, notions of feature redundancy are normally in terms of feature correlation and feature-target concept correlation. Redundant features are assembled in a cluster and a representative feature can be taken out of the cluster. As a result, only a very small number of discriminative features are selected.

Correlation Measures

Correlation Measures seek to quantify statistically how closely related variables are. Correlation is a statistical measure that indicates the extent to which two or more variables fluctuate together. A positive correlation indicates the extent to which those variables increase or decrease in parallel; a negative correlation indicates the extent to which one variable increases as the other decreases.

Feature Subset Selection Algorithm

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 can efficiently and effectively deal with both irrelevant and redundant features, and obtain a good feature subset. The achievement of this algorithm 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. Proposed FAST algorithm, 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. In order to more precisely introduce the algorithm, and because proposed feature subset selection framework involves irrelevant feature removal and redundant feature elimination,

Minimum Spanning Tree Construction

Given a connected, undirected graph, a spanning tree of that graph is asub graph that is a tree and connects all the vertices together. A single graph can have many different spanning trees and also assign a weight to each edge, which is a number representing how unfavorable it is, and use this to assign a weight to a spanning tree by computing the sum of the weights of the edges in that spanning tree. A minimum spanning tree (MST) or minimum weight spanning tree is then a spanning tree with weight less than or equal to the weight of every other spanning tree. More generally, any undirected graph (not necessarily connected) has a minimum spanning forest, which is a union of minimum spanning trees for its connected components.There are now two algorithms commonly used, Prim's algorithm and Kruskal's algorithm. All three are greedy algorithms that run in polynomial time, so the problem of finding such trees is in FP, and related decision problems such as determining whether a particular edge is in the MST or determining if the minimum total weight exceeds a certain value are in P. Another greedy algorithm not as commonly used is the reverse-delete algorithm, which is the reverse of Kruskal's algorithm.

If the edge weights are integers, then deterministic algorithms are known that solve the problem in $O(m + n)$ integer operations, where m is the number of edges, n is the number of vertices.

Partitioning of Minimum Spanning Tree

A partitions is a set of sets of elements of a set.

- Every element of the set belong to one of the sets in the partition.
- No element of the set belong to more than one of the sub-sets.
- Every element of a set belongs to one and only one of the sets of a partition.

The forest of trees is a partition of the original set of nodes. Initially all the sub-sets have exactly one node in them. As the algorithm progresses, we form a union of two of the trees (sub-sets), until eventually the partition has only one sub-set containing all the nodes.A partition of a set may be thought of as a set of *e*quivalence classes. Each sub-set of the partition contains a set of equivalent elements (the nodes connected into one of the trees of the forest). This notion is the key to the cycle detection algorithm. For each sub-set, we denote one element as the representative of that sub-set or equivalence class. Each element in the sub-set is, somehow, equivalent and represented by the nominated representative. Add elements to a tree, we arrange that all the elements point to their representative. As we form a union of two sets, we simply arrange that the representative of one of the sets now points to any one of the elements of the other set.

So the test for a cycle reduces to: for the two nodes at the ends of the candidate edge, find their representatives. If the two representatives are the same, the two nodes are already in a connected tree and adding this edge would form a cycle. The search for the representative simply follows a chain of links. Each node will need a representative pointer. Initially, each node is its own representative, so the pointer is set to NULL. As the initial pairs of nodes are joined to form a tree, the representative pointer of one of the nodes is made to point to the other, which becomes the representative of the tree. As trees are joined, the representative pointer of the representative of one of them is set to point to any element of the other. (Obviously, representative searches will be somewhat faster if one of the representatives is made to point directly to the other).

Selection of Representative Features

Feature selection is frequently used as a preprocessing step to machine learning. It is a process of choosing a subset of original features so that the feature space is optimally reduced according to a certain evaluation criterion. In recent years, data has become increasingly larger in both number of instances and number of features in many applications such as text categorization, image retrieval. This may cause serious problems to many machine learning algorithms with respect to scalability and learning performance. For example, high dimensional data (i.e., data sets with hundreds or thousands of features) can contain high degree of irrelevant and redundant information which may greatly degrade the performance of learning algorithms. Therefore, feature selection becomes very necessary for machine learning tasks when facing high dimensional data, however this trend on both size and dimensionality also poses severe challenges to feature selection algorithms. Some of the recent research efforts in feature selection have been focused on these challenges from handling a huge number of instances to dealing with high dimensional data.This work is concerned about feature selection for high dimensional data.

Feature selection algorithms fall into two broad categories, the filter model or the wrapper model. The filter model relies on general characteristics of the training data to select some features without involving any learning algorithm. The wrapper model requires one predetermined learning algorithm in feature selection and uses its performance to evaluate and determine which features are selected. As for each new subset of features, the wrapper model needs to learn a hypothesis (or a classifier). It tends to find features better suited to the predetermined learning algorithm resulting in superior learning performance, but it also tends to be more computationally expensive than the filter model . When the number of features becomes very large, the filter model is usually chosen due to its computational efficiency.

4. RESULT ANALYSIS

In our system,we are searching for best accurate results. By giving query to the server,the server will remove irrelevant features and searching for relevant features by using minimum spanning tree.To remove the irrelevant features two algorithms are commonly used, Prim's algorithm and Kruskal's algorithm. All three are greedy algorithms that run in polynomial time, Find out which server is in the shortest path, then take relevant features and then find nearest path of the next server for other relevant features. Finally merge all the relevant features and produce single results.

Generally the individual evaluation-based featureselection algorithms of FAST, FCBF, and Relief F aremuch faster than the subset evaluation based algorithmsof CFS, Consist, and FOCUS-SF. FAST isconsistently faster than all other algorithms. Theruntime of FAST is only 0.1 percent of that of CFS,2.4 percent of that of Consist, 2.8 percent of that ofFOCUS-SF, 7.8 percent of that of ReliefF, and76.5 percent of that of FCBF, respectively.

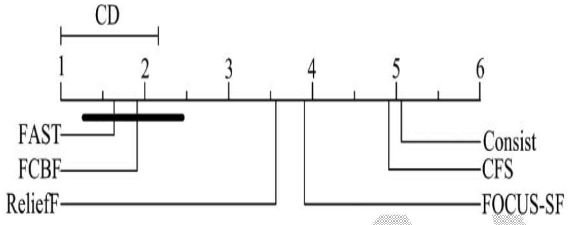


Figure1 Runtime comparison of all feature selection algorithms

5. CONCLUSION

In this paper a clustering-based feature subset selection algorithm for high dimensional data. 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 proposed algorithm, a cluster consists of features. Each cluster is treated as a single feature and thus dimensionality is drastically reduced.Then compare the performance of the proposed algorithm with those of the five well-known feature selection algorithms FCBF, ReliefF, CFS, Consist, and FOCUS-SF on the 35 publicly available Machine Learning, pp. 25-32, 1993. image, microarray, and text data from the four different aspects of the proportion of selected features, runtime, classification accuracy of a given classifier, and the Win/Draw/Loss record. Generally, the proposed algorithm obtained the best proportion of selected features,

REFERENCES

[1] A. Arauzo-Azofra, J.M. Benitez, and J.L. Castro, "A Feature SetMeasure Based on Relief," Proc. Fifth Int'l Conf. Recent Advances inSoft Computing, pp. 104-109, 2004.

[2] L.D. Baker and A.K. McCallum, "Distributional Clustering ofWords for Text Classification," Proc. 21st Ann. Int'l ACM SIGIRConf. Research and Development in information Retrieval, pp. 96-103,1998.

[3] R. Battiti, "Using Mutual Information for Selecting Features inSupervised Neural Net Learning," IEEE Trans. Neural Networks,vol. 5, no. 4, pp. 537-550, July 1994.

[4] D.A. Bell and H. Wang, "A Formalism for Relevance and ItsApplication in Feature Subset Selection," Machine Learning, vol. 41,no. 2, pp. 175-195, 2000.

[5] J. Biesiada and W. Duch, "Features Election for High-Dimensionaldata a Pearson Redundancy Based Filter," Advances in SoftComputing, vol. 45, pp. 242-249, 2008.

[6] R. Butterworth, G. Piatetsky-Shapiro, and D.A. Simovici, "OnFeature Selection through Clustering," Proc. IEEE Fifth Int'l Conf.Data Mining, pp. 581-584, 2005.

[7] C. Cardie, "Using Decision Trees to Improve Case-Based Learning,"Proc. 10th Int'l Conf.