Extraction of Features from Multidimensional Data using Subset Selection Algorithm

Gelli Archana[1], K.Nagamani[2]

P.G. Scholar, Dept. of CSE, Krishnaveni Engineering College for Women, Narasaraopet, Andhra Pradesh, India[1]
Asst Professor, Krishnaveni Engineering College for Women, Narasaraopet, Andhra Pradesh, India [2]
ksj.archana@gmail.com, knmani111@gmail.com

Abstract: *Feature Extraction 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. Based on these criteria, a fast clustering-based feature selection algorithm, FAST, is proposed and experimentally evaluated in this paper. The FAST algorithm works in two steps. In the first step, features are divided into clusters by using graph-theoretic clustering methods. 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.*

Keywords: *Feature Extraction, Minimum Spanning Tree, Clustering*

1. INTRODUCTION

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. Many feature subset selection methods have been proposed and studied for machine learning applications. They can be divided into four broad categories: the Embedded, Wrapper, Filter, and Hybrid approaches.

1.1. Existing System

The embedded methods incorporate feature selection as a part of the training process and are usually specific to given learning algorithms, and therefore may be more efficient than the other three categories. Traditional machine learning algorithms like decision trees or artificial neural networks are examples of embedded approaches. The wrapper methods use the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. However, the generality of the selected features is limited and the computational complexity is large. The filter methods are independent of learning algorithms, with good generality. Their computational complexity is low, but the accuracy of the learning algorithms is not guaranteed. The hybrid methods are a combination of filter and wrapper methods by using a filter method to reduce search space that will be considered by the subsequent wrapper. They mainly focus on combining filter and wrapper methods to achieve the best possible performance with a particular learning algorithm with similar time complexity of the filter methods.

1.2. Proposed System

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 feature(s). 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. Our proposed FAST algorithm 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 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 correlated features are likely both to be 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 identify redundant features.

1.3. Framework of Feature Extraction

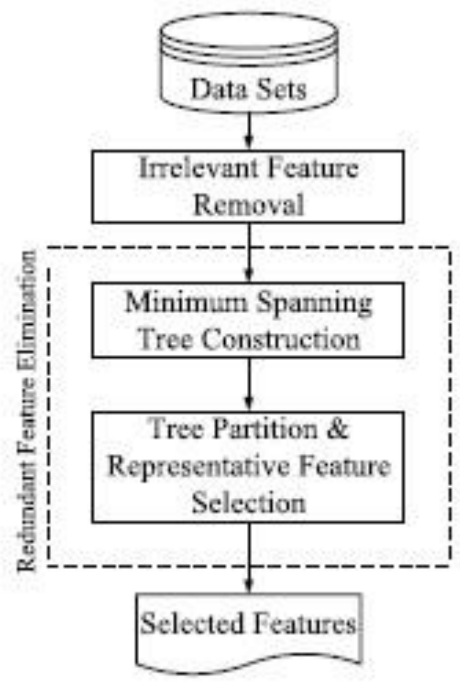
Data Sets
Irrelevant Feature Removal
Minimum Spanning Tree Construction
Tree Partition & Representative Feature Selection
Redundant Feature Elimination
Selected Features

Fig. *Framework of Feature Cluster-Based Extraction Algorithm*

2. Feature Cluster Based Extraction 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 the class, yet uncorrelated with each other.” Keeping these in mind, we develop a novel 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 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, 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. Feature subset selection can be the process that identifies and retains the strong T-Relevance features and selects R-Features from feature clusters. The behind heuristics are that

- Irrelevant features have no/weak correlation with target concept;
- Redundant features are assembled in a cluster and a representative feature can be taken out of the cluster.

3. Algorithm and Time Complexity Analysis

The proposed FAST algorithm logically consists of three steps: (i) removing irrelevant features, (ii) constructing a MST from relative ones, and (iii) partitioning the MST and selecting representative features.

3.1. Algorithm: FAST

Inputs: $D(F_1, F_2, \ldots, F_m, C)$ - the given data set θ - the $T - Relevance\ threshold.$

Output: $S - selected\ feature\ subset.$

```
for i = 1 to m do
    $T - Relevance = SU(F_i, C)$
    if $T - Relevance > \theta$ then
        $S = S \cup \{F_i\}$;
$G = NULL$;
for each pair of features $\{F_i', F_j'\} \subset S$ do
    $F - Correlation = SU(F_i', F_j')$
    Add $F_i' \frac{and}{or} F_j'$ to $G$ with $F$ Correlation
  as the weight of the corresponding edge
$minSpanTree = Prim\ (G)$;
$Forest = minSpanTree$
for each edge $E_{ij} \in Forest$ do
    if $SU(F_i', F_j') < SU(F_i', C) \wedge SU(F_i', F_j') < SU(F_j', C)$ then
        $Forest = Forest - E_{ij}$
$S = \emptyset$
for each tree $T_i \in Forest$ do
    $F_R^j = argmax_{F_k' \in T_i} SU\ (F_k', C)$
    $S = S \cup \{F_R^j\}$;
return $S$
```

3.2. Time Complexity

The major amount of work for Algorithm 1 involves the computation of *SU* values for *T*-Relevance and *F*-Correlation, which has linear complexity in terms of the number of instances in a given data set.

The first part of the algorithm has a linear time complexity $O(m)$ in terms of the number of features *m*. Assuming $k(1 \leq k \leq m)$ features are selected as relevant ones in the first part, when $k = 1$, only on feature is selected. The second part of the algorithm firstly constructs a complete graph from relevant features and the complexity is $O(k^2)$, and then generates a MST from the graph using Prim Algorithm whose time complexity is $O(k^2)$. The third part partitions the MST and Chooses the representative features with the complexity of $O(k)$. Thus when $1 < k \leq m$, the complexity of the Algorithm is $O(m)$

4. Conclusion

This project presented a novel clustering – based feature extraction 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 extracting representative features.

The purpose of cluster analysis has been established to be more effective than feature extraction algorithms. Since high dimensionality and accuracy are the two major concerns of clustering, we have considered them together in this paper for the finer cluster for removing the irrelevant and redundant features. The proposed supervised clustering algorithm is processed for high dimensional data to improve the accuracy and check the probability of the patterns. Retrieval of relevant data should be faster and more accurate. This displays results based on the high probability density thereby reducing the dimensionality of the data.

Future Enhancement

In the near feature, we plan to analyze the distinct types of relationship measures and some formal properties of feature space.

References

[1] Almuallim H. and Dietterich T.G., Algorithms for Identifying Relevant Features, In Proceedings of the 9th Canadian Conference on AI, pp 38-45, 1992.

[2] Arauzo-Azofra A., Benitez J.M. and Castro J.L., A feature set measure based on relief, In Proceedings of the fifth international conference on Recent Advances in Soft Computing, pp 104-109, 2004.

[3] Bell D.A. and Wang, H., A formalism for relevance and its application in feature subset selection, Machine Learning, 41(2), pp 175-195, 2000.

[4] Biesiada J. and Duch W., Features election for high-dimensionaldatała Pearson redundancy based filter, AdvancesinSoftComputing, 45, pp 242C249, 2008.

[5] Butterworth R., Piatetsky-Shapiro G. and Simovici D.A., On Feature Selection through Clustering, In Proceedings of the Fifth IEEE international Conference on Data Mining, pp 581-584, 2005.

[6] Cardie, C., Using decision trees to improve case-based learning, In Proceedings of Tenth International Conference on Machine Learning, pp 25-32, 1993.

[7] Butterworth R., Piatetsky-Shapiro G. and Simovici D.A., On Feature Selection through Clustering, In Proceedings of the Fifth IEEE international Conference on Data Mining, pp 581-584, 2005.

Authors' Biography

Gelli Archana received the B.Tech degree in Information Technology in the year 2012 and pursuing M.Tech degree in Computer Science and Engineering from Krishnaveni Engineering College for Women.

K.Nagamani received her M.Tech degree in Computer Science and MCA Degree. She is currently working as an Asst Professor in Krishnaveni Engineering College for Women.