International Journal of Electrical Electronics & Computer Science Engineering
Special Issue - EEECOS 2014, ISSN : 2348 2273
Available Online at www.ijeecse.com

Efficient Feature Subsets Selections using Suffix Tree Clustering Schemes

K. Pradeep Kumar,Y. C. Ashok Kumar, K. Venkateswara Rao.
Department of Computer Science and Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada.
ycashokkumar@gmail.com,kuthadi.560@gmail.com

Abstract: Feature selection is one of the important features of data mining that concerns with exploring and retrieving a subset of data with most useful features and shares a relevancy with the original entire data set of features. Various feature selection were developed previously and their performance is validated solely on efficiency and effectiveness in generating subset results. While the efficiency factor involves the processing time required to find the feature subsets, the effectiveness factor is concerned with the quality of the feature subsets. Based on these parameters, Earlier a fast clustering-based feature selection algorithm (FAST) with a minimum-spanning tree (MST) implementation at its core is proposed and experimentally evaluated. We propose to use the Suffix tree schemes. The scheme initially involves the original features to be categorized into clusters by using graph based clustering solutions. Then the most representative feature that strongly relates with the target classes is identified from each cluster to form the feature subsets. Features formulated using the Suffix tree based FAST clustering strategy has the highest probability of producing feature subsets of useful and independent features and its performance are evaluated through an empirical study. Experimental results validate our claim.

Keywords: Classifiers, Data Mining, Feature Clustering, Feature Subset Selection, Filter method, Graph-based clustering.

I. INTRODUCTION

With the respect to choosing a good feature subsets in regard to certain target concepts, feature subset selections presents an effective alternative way for reducing data dimensionalities, removing in cohesive and irrelevant data, improving learning accuracies and comprehensibility of the outputs. Many feature subset extraction methods have been proposed earlier and studied vigorously for various machine learning applications. The embedded methods include feature selection as part of the training process itself and are usually constrained to a given learning algorithms, and hence therefore may be more effective than the other prevalent methods. Wrapper methods on the other hand uses the predictive accuracy of a pre fixed learning algorithms to determine validity of the selected sub-sets, the accuracy and efficiency of these learning algorithms is usually high and moderate.

Clustering in data mining can be regarded as an unsupervised learning mechanism since it mainly deals with finding a structure or relevance's in a pool of unlabeled data, which share certain similarities between them and also are dissimilar to the data objects belonging to other neighboring adjacent clusters. Clustering is used to identify groups of related data tuples that are used as a starting point for exploring further subtle relationships within the data sets. A fast supervised attribute clustering algorithm is proposed to improve the accuracy, extraction and efficiency of feature subsets.

With respect to many filter feature selection approaches, our application of cluster analysis has been proven to be more effective than prior feature selection algorithms. In the second phase, the most representative and recurring feature that can strongly relate to pre fixed target classes is selected from each cluster to form the final feature subsets. In our pursuit, we apply graph centric clustering methods to formulate feature subsets.

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,
- Irrelevant features provide no useful information in any context.

Feature selection is also useful as part of the data analysis process, as shows which features are important for prediction, and how these features are related. Feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy, and improving result comprehensibility.

II. RELATED

Many feature subset selection methods have been proposed and studied for machine learning applications. They can be divided into four broad categories:

Embedded: Incorporate feature selection as a part of the training process and are usually specific to given learning algorithms like decision trees or artificial neural networks.

- Pros: High Efficiency
- Cons: Generality of the selected features is limited.

Wrapper: Uses the predictive accuracy of a predetermined learning algorithm to determine the goodness of the selected subsets.

- Pros: High Accuracy
- Cons: Computational complexity is large.

Filter: Determines the validity of the selected subsets and are independent of learning algorithms.

- Pros: Good Generality, Low Computational Complexity.
- Cons: Lacks in accuracy due to lack of learning algorithms.

Hybrid: Uses a Combination of filter and wrapper methods. By using a filter method we 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.

- Pros: Handles Large Features. Supports Generality
- Cons: Computationally Expensive, Over fits on Small Training Sets

Fast Clustering Algorithm: Feature selection involves extracting a feature from the data cluster. The Fast clustering driven feature selection algorithm divides the features in a cluster using graph implementation methods [2]. This sort of partitioning the vertexes in a large graph structure into different clusters and thus differentiating between densely and sparsely connected data graphs into manageable structures. To achieve and attain the efficiency of fast clustering, MST for prior approaches which will bw replaced with Suffix tree a implementations were used for easy retrieval of features [1]. It also improves the performance of classifier further.

Discordant Information Features: The strategies implemented so far for text classification, categorization and word clustering as prescribed in [1] has proven to be vital. Distributed clustering techniques applied to a homogenous data scenario fail to reduce dimensionalities. In order to overcome this problem previously Feature Isolation algorithms were used for raising the value of features called weights and later assess the functionality and effectiveness of the classifier[1]. The clustering method's complexity will be reduced and accuracy is increased due to usage of support vector machine concepts. The technique involves defining data over a vector space where the classification difficulty arises in locating the decision surface. It filters the data points of one class from the other's. In the event of a large linearly separable data, the choice surface happens to be a hyper plane that maximizes the margin between two classes. Isolation clustering algorithms can encounter two major problems. Initially it involves the problem of determining which cluster must be split then the problem of how to implement the splitting procedures of the selected cluster.

Mutual Information: It is used for features selection procedures through neural network based learning using MIFS Algorithm prescribed in [2].It involves accessing the information tuples from complex classification procedures. To overcome this complexity MIFS algorithm implements redundant data, irrelevant features and dimensionality removal/reduction procedures [3]. This method also maintains the feedbacks to correct anomalies in learning using back propagation algorithms [4]. Greedy Heuristic techniques evaluates the data content from each individual feature, within that each feature is the starting data point of a pruning algorithm. This pruning algorithm considers a subset of features from an initial pool of available features. Mutual information estimator bridges missing data and then later uses it to attain feature subset selections.

Feature selection using clustering principles: The paper [3] suggests methods for feature selection using clustering concepts. This kind of selection process involving selecting a combination of features from relevant data points. Using the feature selection technique using this scenario results in many redundant and irrelevant data. [4] Redundant or duplicate or irrelevant features, which provide no more new and useful information [1] than the already selected features. Feature selection method is a subset of the field of feature extraction. Feature extractions creates and explores new features from the original data sets of numerous diverse features, this further returns a subset of the features. Feature selection increases models interpretability through shorter training times and thereby minimizing over fitting. One form of wrapper methods used a specific predictive model to achieve feature subsets categorization [1]. Each and every formulated new subset is used to devise a better model, which is later tested on a data holdout sample. Like wrapper methods training a best model for each subset is very computationally expensive and rigorous process but it is inevitable since it usually provides better performing feature set for that particular model.

Bi-normal Division: This technique mainly concentrates on text classification[3], feature selection is to make huge problems computationally efficient, and storage resources for each and every feature make use of classifiers. To overcome the problem, filter method can be used to remove the irrelevant features and produce a feature set which is not tuned to an exact type of predictive model [1].

Analysis of Relief Functions: Algorithm adopted for attribute estimators to identify conditional dependencies among attributes and afford attribute estimation in classification. Feature selection is the main problem. To overcome this removes the redundant feature from collecting clusters using relief.

III. PROPOSED SCHEME

Unsupervised Graph Theoretic Clustered Learning: This algorithm is used to explore clusters that have a high probable density of relevant items and also an iterative class refinement will further increase the efficiency of clustering process. Let X represents the pool of attributes of the original data tuples, while P and Q are the set of actual and augmented attribute sets, respectively, selected by the proposed attribute clustering algorithm. Let U is the unrefined cluster related with the attribute V and W, the finer cluster of V, represents the set of attributes of U which are merged and averaged with the attribute V to formulate the augmented cluster representative V.

International Journal of Electrical Electronics & Computer Science Engineering
Special Issue - EEECOS 2014, ISSN : 2348 2273
Available Online at www.ijeecse.com

Suffix Tree: A Suffix Tree is a data structure that keeps track of all n-grams of any length in a set of word strings, while allowing strings to be inserted incrementally in time linear to the number of words in each string. The algorithm is theoretically fast, with a runtime of 0(n), where n is the total number of words in all combined document snippets. The algorithm has the important characteristic that the outputted clusters can have overlapping documents. This has the advantage that it ensures that a large number of substantial clusters can be generated, each of which can be labeled fairly accurately. From a user point of view, however, this feature has the disadvantage in that it increases the number of possible document listings that the user may need to look through. This is a drawback that is not easily factored into traditional evaluation techniques. The algorithm we have has several steps:

Preparing the Documents:

- Retrieving the data tuples and parsing and stemming the results.

Suffix Tree Construction

- Inserting the strings associated with each document onto the suffix tree.

Merging Clusters

- Combining similar nodes of the suffix tree.

Labeling Clusters

- Generating a label for each cluster.

Scoring Clusters

- Ranking clusters.

Algorithm:

```
Update( new_suffix )
{
  current_suffix = active_point
  test_char = last_char in new_suffix
  done = false;
  while ( !done ) {
        if current_suffix ends at an explicit node {
          if the node has no descendant edge starting with test_char
                create new leaf edge starting at the explicit node
          else
                done = true;
        } else {
          if the implicit node's next char isn't test_char {
                split the edge at the implicit node
                create new leaf edge starting at the split in the edge
          } else
                done = true;
        }
        if current_suffix is the empty string
          done = true;
        else
           current_suffix = next_smaller_suffix( current_suffix )
  }
  active_point = current_suffix
}
```

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.

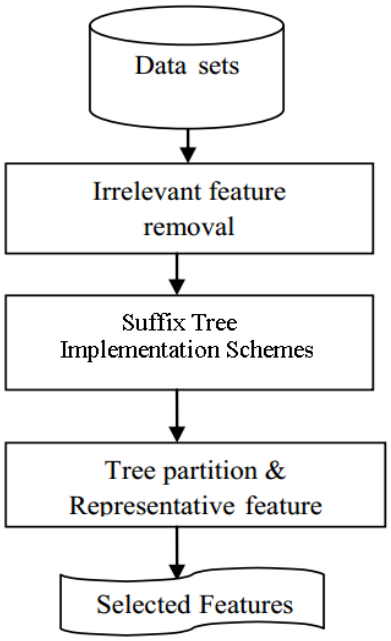

Fig. 1: System Architecture

The major amount of work for FAST Algorithm involves the computation of values for T-Relevance and F-Correlation, which has linear complexity in terms of the number of instances in a given data set. This requires high computational cost especially if we implement correlation measures between different data types in a feature space. To get a useful clustering data according to correlation measures we require the following:

1) The number of clusters should be variable and not fixed a priori to the clustering

2) The diameter of each cluster should not exceed a specific amount.

The clustering algorithm that best satisfies both requirements is the Suffix Tree clustering algorithm.

Suffix Tree is an alternative method of partitioning data in some cases better than MST. It doesn't require more computing power than MST and always returns the same result when run several times. Reduces the computational cost involved in the generation clustering information.

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.

IV. CONCLUSION AND FUTUREWORK

Time and again the efficiency of cluster analysis has been established in various mining implementations and it's effectiveness in feature selection algorithms is recent innovation. Since high dimensional tuples and their classification accuracies for different data are some major concerns of clustering implementations, For removing the irrelevant and redundant features we proposed to use the fast clustering algorithm with suffix tree implementations. The proposed clustering algorithm will process, filter the high dimensional data to achieve efficient and accurate probabilistic cluster patterns. Retrieval of relevant data is faster and more accurate compared to any of the prior approaches. Implementation fuzzy C-means clustering mechanism which happens to be a more recent and efficient one can be explored as a future work.

V. REFERENCES

[1] R. Battiti, "Using Mutual Information for Selecting Features in Supervised Neural Net Learning," July1994.

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

[3] Qinbao song,Jingjie Ni and Guangtao Wang, "A Fast Clustering Based Feature Subset Selection Algorithm for High-Dimensional Data"IEEE Trans.,vol 25,no.1,January 2013.

[4] I.S. Dhillon, S. Mallela, and R. Kumar, "A Divisive Information Theoretic Featur e Clustering Algorithm for Text Classification, "J. Machine Learning Research, vol. 3, pp. 1265-1287, 2003.

[5] M. Robnik-Sikonja and I. Kononenko, "Theoretical and Empirical Analysis of Relief and ReliefF," Machine Learning, vol. 53, pp. 23-69, 2003.

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

[7] G. Forman, "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," J. Machine Learning Research, vol. 3,pp. 1289-1305, 2003.