International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 07
June 2019

A Novel approach for Clustering Based Feature Data Selection Technique Algorithm for High Dimensional Data

Mr. Amos R[1], Mr. Kowshik N[2],Ms. Suraksha M S[3]
[1]Assistant Professor, Department of MCA, MIT Mysore, seeamos@gmail.com
[2] PG Research Scholar, Department of MCA, MIT Mysore, kowshikurs75@gmail.com
[3] PG Research Scholar, Department of MCA, MIT Mysore, m.s.surakshashekhar@gmail.com

Abstract—*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. 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. The efficiency and effectiveness of the FAST algorithm are evaluated through an empirical study. Extensive experiments are carried out to compare FAST and several representative feature selection algorithms, namely, FCBF, ReliefF, CFS, Consist, and FOCUS-SF, with respect to four types of well-known classifiers, namely, the probability-based Naive Bayes, the tree-based C4.5, the instance-based IB1, and the rule-based RIPPER before and after feature selection. The results, on 35 publicly available real-world high dimensional image, microarray, and text data, demonstrate that FAST not only produces smaller subsets of features but also improves the performances of the four types of classifiers.*

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

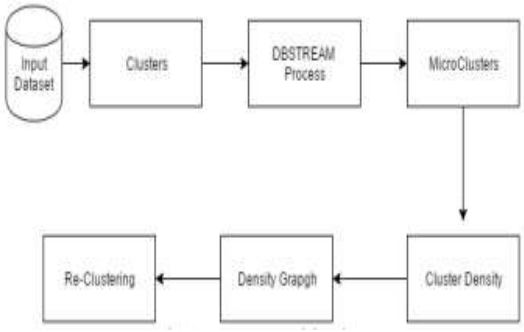
Introduction:

Clustering is a semi-supervised learning problem, which tries to group a set of points into clusters such that points in the same cluster are more similar to each other than points in different clusters, under a particular similarity matrix. Feature subset selection can be viewed as the process of identifying and removing as many irrelevant and redundant features as possible. This is because 1) irrelevant features do not contribute to the predictive accuracy, and 2) redundant features do not redound to getting a better predictor for that they provide mostly information which is already present in other feature(s).

System architecture:

The figure illustrates the system flow of the existing framework of automatic clustering on Density Metrics, which consists of the following steps. First we calculate the object density and density based distance. And then find clusters with their centers.

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 07
June 2019



EXISTING METHOD

The embedded techniques 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. Old machine learning algorithms like decision trees or artificial neural networks are examples of embedded methods. 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 generally high. However, the generality of the selected features is limited and the computational complexity is large. The next is filter methods which are independent of learning algorithms, with good generality. Their computational complexity is lower than previous one, but the accuracy of the learning algorithms is not guaranteed. The last methods known as 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.

PROPOSED METHOD

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. Advantages: 1. Good feature subsets contain features highly correlated with (predictive of) the class, yet uncorrelated with each other. 2. The efficiently and effectively deal with both irrelevant and redundant features, and obtain a good feature subset.

DISTRIBUTED CLUSTERING The Distributional clustering has been 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 associated with each word by Baker and McCallum . As distributional clustering of words are agglomerative in nature, and result in suboptimal word clusters and high

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 07
June 2019

computational cost, proposed a new information-theoretic divisive algorithm for word clustering and applied it to text classification. Unfortunately, the cluster evaluation measure based on distance does not identify a feature subset that allows the classifiers to improve their original performance accuracy. Furthermore, even compared with other feature selection methods, the obtained accuracy is lower.

SUBSET SELECTION ALGORITHM The 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. 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

Distance-based Clustering Techniques Distance-based algorithms analyze the dissimilarity between samples by means of a distance metric and assess the most representative pattern of each cluster, called centroid. Afterwards, the class is decided by assigning the sample to the closest centroids are found targeting small dissimilarity distances to the samples of the own cluster and large dissimilarity distances to the samples of the other clusters. Obviously, there are situations when it becomes unclear how to assign a distance measure to a data set and how to associate the weights of the features.

Algorithm 1 Update DBSTREAM clustering.

Require: Clustering data structures initially empty or 0
$\mathcal{MC}$ ▷ set of MCs
$mc \in \mathcal{MC}$ has elements $mc = (\mathbf{c}, w, t)$ ▷ center, weight, last update time
$\mathbf{S}$ ▷ weighted adjacency list for shared density graph
$s_{ij} \in \mathbf{S}$ has an additional field t ▷ time of last update
t ▷ current time step

Require: User-specified parameters
r ▷ clustering threshold
λ ▷ fading factor
t_{gap} ▷ cleanup interval
w_{min} ▷ minimum weight
α ▷ intersection factor

function UPDATE($\mathbf{x}$) ▷ new data point $\mathbf{x}$
  $\mathcal{N} \leftarrow$ findFixedRadiusNN($\mathbf{x}, \mathcal{MC}, r$)
  if $|\mathcal{N}| < 1$ **then** ▷ create new MC
    add $(\mathbf{c} = \mathbf{x}, t = t, w = 1)$ to $\mathcal{MC}$
  else ▷ update existing MCs
    for each $i \in \mathcal{N}$ **do**
      $mc_i[w] \leftarrow mc_i[w]\, 2^{-\lambda(t - mc_i[t])} + 1$
      $mc_i[\mathbf{c}] \leftarrow mc_i[\mathbf{c}] + h(\mathbf{x}, mc_i[\mathbf{c}])(\mathbf{x} - mc_i[\mathbf{c}])$
      $mc_i[t] \leftarrow t$
      ▷ update shared density
      for each $j \in \mathcal{N}$ where $j > i$ **do**
        $s_{ij} \leftarrow s_{ij}\, 2^{-\lambda(t - s_{ij}[t])} + 1$
        $s_{ij}[t] \leftarrow t$
      end for
    end for
    ▷ prevent collapsing clusters
    for each $(i, j) \in \mathcal{N} \times \mathcal{N}$ and $j > i$ **do**
      if dist$(mc_i[\mathbf{c}], mc_j[\mathbf{c}]) < r$ **then**
        revert $mc_i[\mathbf{c}], mc_j[\mathbf{c}]$ to previous positions
      end if
    end for
  end if
  $t \leftarrow t + 1$
end function

Algorithm 2 Cleanup process to remove inactive micro-clusters and shared density entries from memory.

Require: λ, α, t_{gap}, t, $\mathcal{MC}$ and $\mathbf{S}$ from the clustering.
function CLEANUP( )
  $w_{weak} = 2^{-\lambda t_{gap}}$
  for each $mc \in \mathcal{MC}$ **do**
    if $mc[w]\, 2^{-\lambda(t - mc[t])} < w_{weak}$ **then**
      remove weak mc from $\mathcal{MC}$
    end if
  end for
  for each $s_{ij} \in \mathbf{S}$ **do**
    if $s_{ij}\, 2^{-\lambda(t - s_{ij}[t])} < \alpha w_{weak}$ **then**
      remove weak shared density s_{ij} from $\mathbf{S}$
    end if
  end for
end function

Algoritham:

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 07
June 2019

Algorithm 3 Reclustering using shared density graph.

Require: λ, α, $w_{\min}$, t, $\mathcal{MC}$ and $\mathbf{S}$ from the clustering.

1: **function** RECLUSTER( )
2: weighted adjacency list $\mathbf{C} \leftarrow \emptyset$ ▷ connectivity graph
3: **for each** $s_{ij} \in \mathbf{S}$ **do** ▷ construct connectivity graph
4: **if** $\mathcal{MC}_i[w] \geq w_{\min} \wedge \mathcal{MC}_j[w] \geq w_{\min}$ **then**
5: $c_{ij} \leftarrow \frac{s_{ij}}{(\mathcal{MC}_i[w]+\mathcal{MC}_j[w])/2}$
6: **end if**
7: **end for**
8: **return** findConnectedComponents($\mathbf{C} \geq \alpha$)
9: **end function**

Types of clustering algorithms

Since the task of clustering is subjective, the means that can be used for achieving this goal are plenty. Every methodology follows a different set of rules for defining the '*similarity'* among data points. In fact, there are more than 100 clustering algorithms known. But few of the algorithms are used popularly, let's look at them in detail:

- **Connectivity models:** As the name suggests, these models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away. These models can follow two approaches. In the first approach, they start with classifying all data points into separate clusters & then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lacks scalability for handling big datasets. Examples of these models are hierarchical clustering algorithm and its variants.

- **Centroid models:** These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a popular algorithm that falls into this category. In these models, the no. of clusters required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These models run iteratively to find the local optima.

- **Distribution models:** These clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution (For example: Normal, Gaussian). These models often suffer from overfitting. A popular example of these models is Expectation-maximization algorithm which uses multivariate normal distributions.

- **Density Models:** These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster. Popular examples of density models are DBSCAN and OPTICS.

Now I will be taking you through two of the most popular clustering algorithms in detail – K Means clustering and Hierarchical clustering. Let's begin.

Major Clustering Approaches

1. Partitioning algorithms: Construct various partitions and then evaluate them by some criterion
2. Hierarchy algorithms: Create a hierarchical decomposition of the set of data (or objects) using some criterion
3. Density-based: based on connectivity and density functions

4. Grid-based: based on a multiple-level granularity structure
5. Model-based: A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other

Conclusion:

CONCLUSION In this paper, we have presented a novel 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. We have compared 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 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, the best runtime, and the best classification accuracy for Naive Bayes, C4.5, and RIPPER, and the second best classification accuracy for IB1. The Win/Draw/Loss records confirmed the conclusions.

REFERENCES

[1] Qinbao Song, Jingjie Ni and Guangtao Wang, “A Fast clustering based feature subset selection algorithm for high dimensional data”, In proceedings of the IEEE Transactions n Knowledge and data engineering, 2013.

[2] L. Yu and H. Liu, “Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution,” Proc. 20th Int’l Conf. Machine Leaning, vol. 20, no. 2, pp. 856-863, 2003.

[3] 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.

[4] A New Clustering Based Algorithm for Feature Subset Selection (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5272-5275.

[5] Dingcheng Feng, Feng Chen_, and Wenli Xu “Efficient Leave-One-Out Strategy for Supervised Feature Selection” TSINGHUA SCIENCE AND TECHNOLOGY ISSN 1007-0214 09/10 pp629 635 Volume 18, Number 6, December 2013.

[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] H. Liu, H. Motoda, and L. Yu, "Selective Sampling Approach to Active Feature Selection," Artificial Intelligence, vol. 159, nos. 1/2, pp. 49-74, 2004.

[8] Qinbao Song, Jingjie Ni, and Guangtao Wang, "A Fast Clustering-Based Feature Subset Selection Algorithm for High-Dimensional Data," IEEE Transaction on Knowledge and Data, Engineering, Vol. 25, No. 1, January 2013.

[9] 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.

[10] Jesna Jose,"Fast for Feature Subset Selection Over Dataset" International Journal of Science and Research (IJSR), Volume 3 Issue 6, June 2014.