Implementation of Fast Clustering Based Feature Subset Selection Algorithm for HDD

T.Gayathri[1]
B.Tech/IT
Prathyusha Institute Of Technology and Management
Chennai
gayusureka@gmail.com

D.Suvidha[2]
B.Tech/IT
Prathyusha Institute of Technology and Management
Chennai
ammushanthi2708@gmail.com

P.V.Monisha[3]
B.Tech/IT
Prathyusha Institute of Technology and Management
Chennai
monisha.p.v@gmail.com

U.JothiLakshmi.[4]
Assistant Professor
Prathyusha Institute of Technolgy and Management
Chennai
jothi.lakshmiu@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. 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.*

Keywords— Feature subset selection, filter method, feature clustering, graph-based 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.

In cluster analysis, graph-theoretic methods have been well studied and used in many applications. Their results have, sometimes, the best agreement with human performance. 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. In our study, we apply graph-theoretic clustering methods to features.

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. In particular, we adopt the minimum spanning tree (MST)-based clustering algorithms, because they do not assume that data points are grouped around centers or separated by a regular geometric curve and have been widely used in practice. Based on the MST method, we propose a Fast clusteringbAsed feature Selection algoriThm (FAST). 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 the final subset of features. Features in different clusters are relatively independent, the clusteringbased strategy of FAST has a high probability of producing a subset of useful and independent features. The proposed feature subset selection algorithm FAST was tested upon 35 publicly available image, microarray, and text data sets.

The experimental results show that, compared with other five different types of feature subset selection algorithms, the proposed algorithm not only reduces the number of features, but also improves the performances of the four well-known different types of classifiers., we describe the related works.

1.2 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. 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 area 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.

The generality of the selected features limited and the computational complexity is large.Their computational complexity is low, but the accuracy of the learning algorithm is not guarantee.

1.3 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.

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

4. FEATURE SUBSET SELECTION ALGORITHM

4.1 Framework and Definitions

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. We achieve this through a new feature selection framework (shown in Fig. 1) 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.

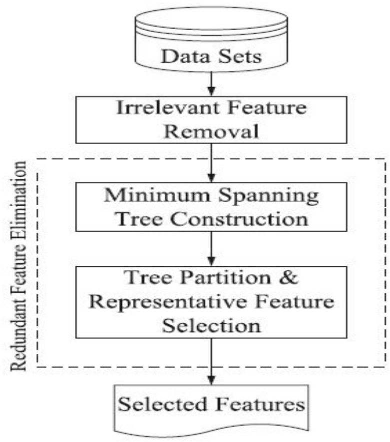

Fig. 1. Frameworkof the proposed feature subset selection algorithm.

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 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 our proposed feature subset selection framework involves irrelevant feature removal and redundant feature elimination, we first present the traditional definitions of relevant and redundant features, then provide our definitions based on variable correlation as follows.

Relevant features have strong correlation with target concept so are always necessary for a best subset, while redundant features are not because their values are completely correlated with each other. Thus, notions of feature redundancy and feature relevance are normally in terms of feature correlation and feature-target concept correlation.

4.2 Algorithm and Analysis

The proposed FAST algorithm logically consists of three steps: 1) removing irrelevant features,2) constructing an MST from relative ones, and 3) partitioning the MST and selecting representative features.

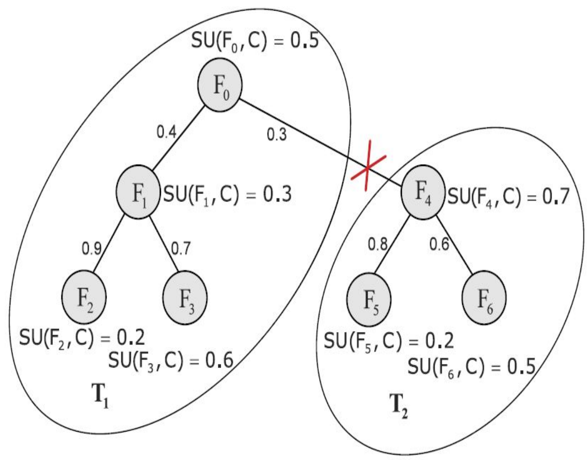

Fig. 2. Example of the clustering step.

In order to cluster the features, we first traverse all the six edges, and then decide to remove the edge (F0; F4) because its weight SU(F0; F4) = 0:3 is smaller than both SU(F0; C) = 0:5 and SU(F4; C) -=0:7. This makes the MST is clustered into two clusters denoted as V (T1) and V (T2). Each cluster is an MST as well. Take V (T1) as an example. From Fig. 2, we know that SU(F0; F1) > SU(F1; C), SU(F1; F2) > SU(F1; C) ^ SU(F1; F2) > SU(F2; C), SU(F1; F3) > SU(F1; C) ^ SU(F1; F3) > SU(F3; C). We also observed that there is no edge exists between F0 and F2, F0 and F3, and F2 and F3. Considering that T1 is an MST, so the SUðF0; F2Þ is greater than SU(F0; F1) and SU(F1; F2), SU(F0; F3) is greater thanSU(F0; F1) and SU(F1; F3), and SU(F2; F3) is greater thanSU(F1; F2) and SU(F2; F3). Thus, SU(F0; F2) > SU(F0; C)^ SU(F0; F2) > SU(F2; C), SU(F0; F3) > SU(F0; C) ^ SU(F0;F3) > SU(F3; C), and SU(F2; F3)> SU(F2; C) ^ SU(F2;F3)> SU(F3; CÞ)also hold. As the mutual information between any pair (Fi, Fj)(I,j = 0, 1,2, 3 ^ i = j) of F0, F1,F2, and F3 is greater than the mutual information between class C and Fi or Fj, features F0; F1; F2, and F3 are redundant. After removing all the unnecessary edges, a forest Forest is obtained. Each tree Tj 2 Forest represents a cluster that is denoted as V (Tj), which is the vertex set of Tj as well. As illustrated above, the features in each cluster are redundant, so for each cluster V (Tj) we choose a representative feature F^j_R R whose T Relevance SU(F^j_R ,C) is the greatest. All F^j_R (j =1 . . . |Forest|) comprise the final feature subset UF^j_R.

5.EMPIRICAL STUDY

5.1. Data Source

For the purposes of evaluating the performance and effectiveness of our proposed FAST algorithm, verifying whether or not the method is potentially useful in practice, and allowing other researchers to confirm our results, 35 publicly available data sets1 were used. The numbers of features of the 35 data sets vary from 37 to 49,52 with a mean of 7,874. The dimensionality of the 54.3 percent data sets exceed 5,000, of which 28.6 percent data sets have more than 10,000 features. The 35 data sets cover a range of application domains such as text, image and bio microarray data classification.

5.2 Experiment Setup

To evaluate the performance of our proposed FAST algorithmand compare it with other feature selection algorithmsin a fair and reasonable way, we set up our experimentalstudy as follows: The proposed algorithm is compared with five different types of representative feature selection algorithms. They are 1) FCBF 2) ReliefF , 3) CFS , 4) Consist, and 5) FOCUS-SF.

2.Four different types of classification algorithms are employed to classify data sets before and after feature selection. They are 1) the probability-based
Naive Bayes (NB), 2) the tree-based C4.5, 3) the instance-based lazy learning algorithm IB1, and 4) the rule-based RIPPER.

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

1) theproportion of selected features 2) the time to obtain the feature subset, 3) the classification accuracy, and 4) the Win/Draw/Loss record are used. The proportion of selected features is the ratio of the number of features selected by a feature selection algorithm to the original number of features of a data set. The Win/Draw/Loss record presents three values on a given measure, i.e., the numbers of data sets for which our proposed algorithm FAST obtains better, equal, and worse performance than other five feature selection algorithms, respectively. The measure can be the proportion of selected features, the
runtime to obtain a feature subset, and the classification
accuracy, respectively.

5.3 Results and Analysis

In this section, we present the experimental results in terms of the proportion of selected features, the time to obtain the feature subset, the classification accuracy, and the Win/Draw/Loss record.

Data set	Proportion of selected features (%) of					
	FAST	FCBF	CFS	ReliefF	Consist	FOCUS-SF
chess	16.22	21.62	10.81	62.16	81.08	18.92
mfeat-fourier	19.48	49.35	24.68	98.70	15.58	15.58
coil2000	3.49	8.14	11.63	50.00	37.21	1.16
elephant	0.86	3.88	5.60	6.03	0.86	0.86
arrhythmia	2.50	4.64	9.29	50.00	8.93	8.93
fqs-nowe	0.31	2.19	5.63	26.56	4.69	4.69
colon	0.30	0.75	1.35	39.13	0.30	0.30
fbis.wc	0.80	1.45	2.30	0.95	1.75	1.75
AR10P	0.21	1.04	2.12	62.89	0.29	0.29
PIE10P	1.07	1.98	2.52	91.00	0.25	0.25
oh0.wc	0.38	0.88	1.10	0.38	1.82	1.82
oh10.wc	0.34	0.80	0.56	0.40	1.61	1.61
B-cell1	0.52	1.61	1.07	30.49	0.10	0.10
B-cell2	1.66	6.13	3.85	96.87	0.15	0.15
B-cell3	2.06	7.95	4.20	98.24	0.12	0.12
base-hock	0.58	1.27	0.82	0.12	1.19	1.19
TOX-171	0.28	1.41	2.09	64.60	0.19	0.19
tr12.wc	0.16	0.28	0.26	0.59	0.28	0.28
tr23.wc	0.15	0.27	0.19	1.46	0.21	0.21
tr11.wc	0.16	0.25	0.40	0.37	0.31	0.31
embryonal-tumours	0.14	0.03	0.03	13.96	0.03	0.03
leukemia1	0.07	0.03	0.03	41.35	0.03	0.03
leukemia2	0.01	0.41	0.52	60.63	0.08	0.08
tr21.wc	0.10	0.22	0.37	2.04	0.20	0.20
wap.wc	0.20	0.53	0.65	1.10	0.41	0.41
PIX10P	0.15	3.04	2.35	100.00	0.03	0.03
ORL10P	0.30	2.61	2.76	99.97	0.04	0.04
CLL-SUB-111	0.04	0.78	1.23	54.35	0.08	0.08
ohscal.wc	0.34	0.44	0.18	0.03	NA	NA
la2s.wc	0.15	0.33	0.54	0.09	0.37	NA
la1s.wc	0.17	0.35	0.51	0.06	0.34	NA
GCM	0.13	0.42	0.68	79.41	0.06	0.06
SMK-CAN-187	0.13	0.25	NA	14.23	0.06	0.06
new3s.wc	0.10	0.15	NA	0.03	NA	NA
GLA-BRA-180	0.03	0.35	NA	53.06	0.02	0.02
Average(Image)	3.59	10.04	6.68	79.85	3.48	3.48
Average(Microarry)	0.71	2.34	2.50	52.92	0.91	0.91
Average(Text)	2.05	3.25	2.64	10.87	11.46	2.53
Average	1.82	4.27	3.42	42.54	5.44	2.06
Win/Draw/Loss	-	33/0/2	31/0/4	29/1/5	20/2/13	19/2/14

5.4 Proportion of Selected Features

Table 2 records the proportion of selected features of the six feature selection algorithms for each data set. From it we observe that

1. Generally all the six algorithms achieve significant reduction of dimensionality by selecting only a small portion of the original features. The FAST, on
average, obtains the best proportion of selected features of 1.82 percent. The Win/Draw/Loss records show FAST wins other algorithms as well.

2. For image data, the proportion of selected features of each algorithm has an increment compared with the corresponding average proportion of selected
features on the given data sets except Consist has an improvement. This reveals that the five algorithms are not very suitable to choose features for image data compared with for microarray and text data. FAST ranks 3 with the proportion of selected features of 3.59 percent that has a tiny margin of 0.11 percent to the first and second best proportion of selected features 3.48 percent of Consist and FOCUS-SF, and a margin of 76.59 percent to the worst proportion of selected features 79.85 percent of ReliefF.

3. For microarray data, the proportion of selected features has been improved by each of the six
algorithms compared with that on the given data sets. This indicates that the six algorithms work well with microarray data. FAST ranks 1 again with the proportion of selected features of 0.71

percent. Of the six algorithms, only CFS cannot choose features for two data sets whose dimensionalities are 19,994 and 49,152, respectively.

6.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 singlefeature 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.

7.REFERNCE:

1.Hall M.A., Correlation-Based Feature Subset Selection for Machine Learning,Ph.D. dissertation Waikato, New Zealand: Univ. Waikato, 1999.

2.Webb G.I., Multiboosting: A technique for combining boosting and Wagging,Machine Learning, 40(2), pp 159-196, 2000.

3.Yu L. and Liu H., Efficient feature selection via analysis of relevance andredundancy, Journal of Machine Learning Research, 10(5), pp 1205-1224,2004.

4.Demsar J., Statistical comparison of classifiers over multiple data sets, J.Mach. Learn. Res., 7, pp 1-30, 2006.

5.Garcia S and Herrera F., An extension on “Statistical Comparisons of Classifiers over Multiple Data Sets” for all pairwise comparisons, J. Mach.Learn. Res., 9, pp 2677-2694, 2008.

6.G.H. John, R. Kohavi, and K. Pfleger, “Irrelevant Features and the Subset Selection Problem,” Proc. 11th Int’l Conf. Machine Learning, pp. 121-129, 1994.

7.K. Kira and L.A. Rendell, “The Feature Selection Problem:Traditional Methods and a New Algorithm,” Proc. 10th Nat’l Conf. Artificial Intelligence, pp. 129-134, 1992.

8.F. Pereira, N. Tishby, and L. Lee, “Distributional Clustering of English Words,” Proc. 31st Ann. Meeting on Assoc. for Computational Linguistics, pp. 183-190, 1993.

9.W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, Numerical Recipes in C. Cambridge Univ. Press 1988.

10.R.C. Prim, “Shortest Connection Networks and Some Generalizations,” Bell System Technical J., vol. 36, pp. 1389-1401, 1957.

11,J.R. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufman,
1993.