Empirical Evaluation of Feature Subset Selection Based on a Real-World Data Set

Petra Perner[1] and Chid Apte[1]

[1]Institute of Computer Vision and Applied Computer Sciences
Arno-Nitzsche-Str. 45,04277 Leipzig
ibaiperenr@aol.com http://www.ibai-research.de
[2]IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA

Abstract. Selecting the right set of features for classification is one of the most important problems in designing a good classifier. Decision tree induction algorithms such as C4.5 have incorporated in their learning phase an automatic feature selection strategy while some other statistical classification algorithm require the feature subset to be selected in a preprocessing phase. It is well know that correlated and irrelevant features may degrade the performance of the C4.5 algorithm. In our study, we evaluated the influence of feature pre-selection on the prediction accuracy of C4.5 using a real-world data set .We observed that accuracy of the C4.5 classifier can be improved with an appropriate feature pre-selection phase for the learning algorithm.

1 Introduction

Selecting the right set of features for classification is one of the most important problems in designing a good classifier. Very often we don't know a-priori what the relevant features are for a particular classification task. One popular approach to address this issue is to collect as many features as we can prior to the learning and data modeling phase. However, irrelevant or correlated features, if present, may degrade the performance of the classifier. In addition, large feature spaces can sometimes result in overly complex classification models that may not be easy to interpret.

In the emerging area of data mining applications, users of data mining tools are faced with the problem of data sets that are comprised of large numbers of features and instances. Such kinds of data sets are not easy to handle for mining. The mining process can be made easier to perform by focussing on a subset of relevant features while ignoring the other ones. In the feature subset selection problem, a learning algorithm is faced with the problem of selecting some subset of features upon which to focus its attention.

In this paper, we present our study on features subset selection and classification with C4.5 algorithm. In Section 2, we briefly describe the criteria used for feature selection and the feature selection methods. Although, C4.5 has a feature selection strategy included in its learning performance it has been observed that this strategy is not optimal. Correlated and irrelevant attributes may degrade the performance of the induced classifier. Therefore, we use feature subset selection prior to the learning

D.A. Zighed, J. Komorowski, and J. Zytkow (Eds.): PKDD 2000, LNAI 1910, pp. 575-580, 2000.

phase. The CM algorithm selects features based upon their rank ordered *contextual merits* [4].The feature selection strategy used by C4.5 and the CM algorithm are reviewed in Section 2. For our experiments, we used a real data set that includes features extracted from x-ray images which describe defects in a welding seam. It is usually unclear in these applications what the right features are. Therefore, most analyses begin with as many features as one extract from the images. This process as well as the images are described in Section 3.

In Section 4, we describe our results. We show that the prediction accuracy of the C4.5 classifier will improve when provided with a pre-selected feature subset. The results show that the feature subsets created by CM algorithm and the feature subset normally extracted by C4.5 have many features in common. However , the C4.5 selects some features that are never selected by the CM algorithm. We hypothesize that irrelevant features are weeded out by the CM feature selection algorithm while they get selected by the C4.5 algorithm. A comparison of the feature ranking done by the CM algorithm with the ranking of the features done by C4.5 for the first 10 features used by C4.5 shows that there is a big difference. Finally, our experiments also indicate that model complexity does not significantly change for the better or worse when pre-selecting features with CM.

2 Feature Subset Selection Algorithms

According to the quality criteria [8] for feature selection, the model for feature selection can be distinguished into the filter model and the wrapper model [1,7]. The wrapper model attempts to identify the best feature subset for use with a particular algorithm, while the filter approach attempts to assess the merits of features from the data alone. Although the wrapper model can potentially produce the best resulting classifier, it does so by doing an exhaustive search over the entire feature space. Various search strategies have been developed in order to reduce the computation time [9] for wrapper algorithms. The filter approach on the other hand is a greedy search based approach that is computationally not as expensive. The feature selection in C4.5 may be viewed as a filter approach, while the CM algorithm may be viewed as a wrapper approach.

2.1 Feature Selection Done by Decision Tree Induction

Determining the relative importance of a feature is one of the basic tasks during decision tree generation. The most often used criteria for feature selection is information theoretic based, such as the Shannon entropy measure I for a data set. If we subdivide a data set using values of an attribute as separators, we obtain a number of subsets. For each of these subsets we can compute the information value. If the the information value of a subset n is i_n, then the new information value is given by $I_i = q_n i_n$, where q_n is the subset of data points with attribute value n. I_i will be smaller than I, and the difference (I- I_i) is a measure of how well the attribute has discriminated between different classes. The attribute that maximizes this difference is selected.

The measure can also be viewed as a class separability measure. The main drawback of the entropy measure is its sensitivity to the number of attributes values

[11]. Therefore C4.5 uses the gain ratio. However, this measure suffers the drawback that it may choose attributes with very low information content of the attribute itself [2].

C4.5 [10]uses a univariate feature selection strategy. At each level of the tree building process only one attribute, the attribute with the highest values for the selection criteria, is picked out of the set of all attributes. Afterwards the sample set is split into sub-sample sets according to the values of this attribute and the whole procedure is recursively repeated until only samples from one class are in the remaining sample set or until the remaining sample set has no discrimination power anymore and the tree building process stops.

As we can see feature selection is only done at the root node over the entire decision space. After this level, the sample set is split into sub-samples and only the most important feature in the remaining sub-sample set is selected. Geometrically it means, the search for good features is only done in orthogonal decision subspaces, which might not represent the real distributions, beginning after the root node. Thus, unlike statistical feature search strategies [3] this approach is not driven by the evaluation measure for the combinatorial feature subset; it is only driven by the best single feature. This might not lead to an optimal feature subset in terms of classification accuracy.

Decision trees users and researchers have recognized the impact of applying a full set of features to a decision tree building process versus applying only a judiciously chosen subset. It is often the case that the latter produces decision trees with lower classification errors, particularly when the subset has been chosen by a domain expert. Our experiments were intended to evaluate the effect of using multivariate feature selection methods as pre-selection steps to a decision tree building process.

2.2 Contextual Merit Algorithm

For our experiment, we used the contextual merit (CM) algorithm [4]. This algorithm employs a merit function based upon weighted distances between examples which takes into account complete feature correlation's to the instance class. The motivation underlying this approach was to weight features based upon how well they discriminate instances that are close to each other in the Euclidean space and yet belong to different classes. By focusing upon these nearest instances, the context of other attributes is automatically taken into account.

To compute contextual merit, the distance d^{k}_{rs} between values z_{kr} and z_{ks} taken by feature k for examples r and s is used as a basis. For symbolic feature, the inter-example distance is 0 if $z_{kr} = z_{ks}$, and 1 otherwise. For numerical features, the inter-example distance is $\min\left(\frac{z_{kr} - z_{ks}}{t_k}, 1\right)$, where t_k is a threshold for feature k (usually 1/ 2 of the magnitude of range of the feature). The total distance between examples r and s is $D_{rs} = \sum_{k=1}^{N_f} d^{k}_{rs}$, and the contextual merit for a feature f is $M_f = \sum_{r=1}^{N} \sum w^{f}_{rs} d^{f}_{rs}$, where N is the total number of examples, C_r is the set of examples not in the same class as examples r, and $\mathrm{w}^{\mathrm{f}}_{\mathrm{rs}}$ is a weight function chosen so that examples that are close together are given greater influence in determining each features merit. In practice , it has been observed that $\frac{1}{D_{rs}^{2}}$ if *s* is one of *k* nearest neighbors to *r*, and 0 otherwise, provides robust behavior as a weight function. Additionally, using $\log_2 \#\overline{C(r)}$ as the value for *k* has also exhibited robust behavior. This approach to computing and ordering features by their merits has been observed to be very robust, across a wide range of examples.

3 Our Data Set

A detailed description of the data set can be found in [6]. Here we can try to briefly sketch out how the data set was created and what features were used.

The subject of investigation is the in-service inspection of welds in pipes of austenitic steel. The flaws to be looked for in the austenitic welds are longitudinal cracks due to intergranular stress corrosion cracking starting from the inner side of the tube. The radio-graphs are digitized with a spatial resolution of 70 mm and a gray level resolution of 16 bit per pixel. Afterwards they are stored and decomposed into various Regions of Interest (ROI) of 50x50 pixel size. The essential information in the ROIs is described by a set of features which are calculated from various image-processing methods.

The final data set contains 36 parameters collected for every ROI. The data set consists of altogether 1924 ROIs with 1024 extracted from regions of no disturbance, 465 from regions with cracks and 435 from regions with under-cuts.

4 Results

Table 1 illustrates the error rate for the C4.5 classifier when using all features as well as error rates for different feature subsets. The error rate was estimated using cross-validation. The improvement in accuracy is two percent for the pruned case. To interpret this improvement, we use a classification analysis conducted earlier [5], where performance actually peaked and then deteriorated as the number of features was increased. We observe similar behavior in our experiments. Classification error is at its minimum when the feature subset size is 20. This is in contrast to the feature subset size of 28 that C4.5 selects when presented with the entire feature set, with no pre-selection.

It may be argued that it is not worth doing feature subset selection before tree induction since the improvement in prediction accuracy is not so dramatic. However, the importance of an improvement, however small, clearly depends on the requirements of the application for which the classifier is being trained. We further observed (Table 4) that about 67% of the total features are used similarly by CM and C4.5, while about 33% of the features are exclusively selected by CM, and 16% are exclusively selected byC4.5. Table 3 shows that the tree does not necessarily become more compact even if a reduced set of features is used. The tree actually becomes even larger in the case with the best error rate. We therefore cannot draw any useful conclusion about feature set size and its relation to model complexity. We also observe (Table 4) that in comparing the two trees generated by C4.5 with and without CM's pre-selection, the feature used for splitting at the root node changes.

5 **Conclusion**

We have studied the influence of feature subset selection based on a filter and wrapper approach to the performance of C4.5. Our experiment was motivated by the fact that C4.5 uses a non-optimal feature search strategy. We used the CM algorithm for feature subset selection which measures importance of a feature based on a contextual merit function. Our results show that feature subset selection can help to improve the prediction accuracy of the induced classifier. However, it may not lead to more compact trees and the prediction accuracy may not increase dramatically.

The main advantage may be that fewer features required for classification can be important for application such as image interpretation where computational costs for extracting the features may be high and require special purpose hardware. For such domains, feature pre-selection to prune down the feature set size may be a beneficial analysis phase.

Table 1. Error Rate for Different Feature Subsets

	Test=Design		Crossvalidation	
Parameters	unpruned	pruned	unpruned	Pruned
all	0,9356	1,6112	24,961	24,545
10	1,5073	3,7942	29,4332	28,7051
15	1,4033	3,0146	26,365	26,4171
20	1,5073	2,5988	23,7649	22,7769
24	0,9356	1,7152	24,493	23,5049
28	0,9875	1,7152	25,117	24,077

Table 2. Error Rates for Different Sizes Feature Sets

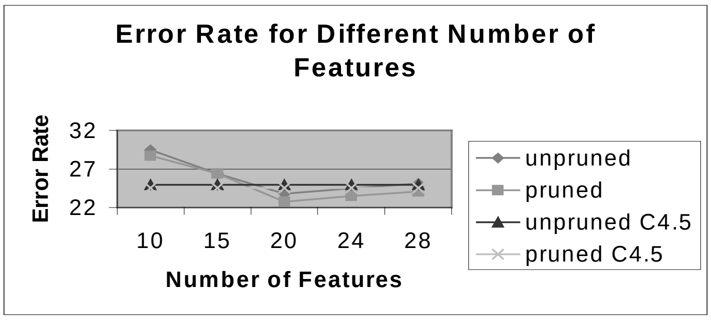
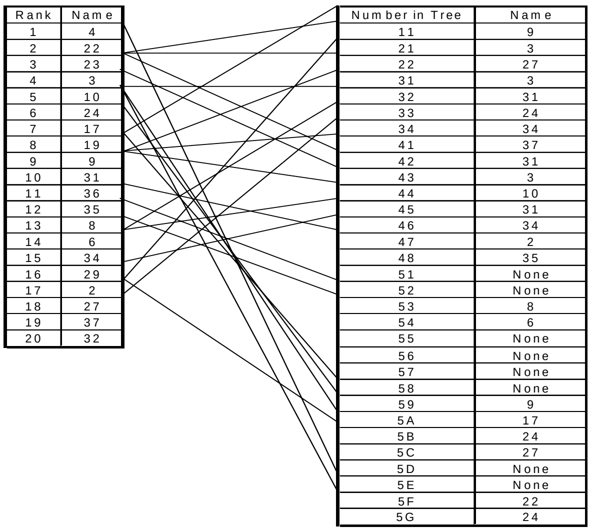


Table 3. Number of Nodes and Edges

Number of Features	10	15	20	24	28	37
Nodes	236	204	178	166	164	161
Edges	237	206	176	137	161	159

Attributes	10	15	20	24	28	C 4.5
2	0	0	1	1	1	1
3	1	1	1	1	1	1
4	1	1	1	1	1	1
5	0	0	0	0	0	1
6	0	1	1	1	1	1
7	0	0		1	1	1
8	0	1	1	1	1	0
9	1	1	1	1	1	1
10	1	1	1	1	1	0
11	0	0	0	0	0	1
12	0	0	0	1	1	1
13	0	0	0	0	0	1
14	0	0	0	0	0	1
15	0	0	0	0	1	1
16	0	0	0	0	0	1
17	0	1	1	1	1	0
18	0	0	0	0	1	1
19	1	1	1	1	1	1
20	0	0	0	1	1	0
21	0	0	0	0	0	1
22	1	1	1	1	1	0
23	0	1	1	1	1	1
24	1	1	1	1	1	1
25	0	0	0	0	0	1
26	0	0	0	0	0	1
27	0	0	1	1	1	1
28	0	0	0	1	1	1
29	0	1	1	1	1	1
30	0	0	0	0	1	1
31	1	1	1	1	1	1
32	0	0	1	1	1	1
33	0	0	0	0	1	1
34	0	0	1	1	1	1
35	1	1	1	1	1	1
36	1	1	1	1	1	1
37	0	0	1	1	1	1
Number used	10	15	20	24	28	31



Rank	Name
1	4
2	22
3	23
4	3
5	10
6	24
7	17
8	19
9	9
10	31
11	36
12	35
13	8
14	6
15	34
16	29
17	2
18	27
19	37
20	32

Number in Tree	Name
11	9
21	3
22	27
31	3
32	31
33	24
34	34
41	37
42	31
43	3
44	10
45	31
46	34
47	2
48	35
51	None
52	None
53	8
54	6
55	None
56	None
57	None
58	None
59	9
5A	17
5B	24
5C	27
5D	None
5E	None
5F	22
5G	24

Table 4 Ranked Feature and the first 10 Features used by Decision Tree

References

[1] T.M Cover, ‚On the possible ordering on the measurement selection problem‘, IEEE Transactions, **SMC-7**(9), 657-661, (1977).

[2] R.Lopez de Mantaras, ‚A distance-based attribute selection measure for decision tree induction‘, *Machine Learning,* **6**(1991), 81-92.

[3] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition*, AcademicPress, 1990.

[4] Se June Hong, ‚Use of contextual information for feature ranking and discretization‘, *IEEE Trans. on Knowledge Discovery and Data Engineering*, (1996).

[5] G. F. Hughes, ‚On the mean accuarcy of statistical pattern recognizers‘, IEEE Transactions, **IT-14**(1), 55-63, (1968).

[6] C. Jacobsen, U.Zscherpel,and P.Perner, *A Comparison between Neural Networks and Decision Trees*, 144-158, Machine Learning and DataMining in Pattern Recognition, Springer Verlag,1999.

[7] R.Kohavi and G.H. John, *The Wrapper Approach*, 33-50, Feature Extraction Construction and Selection, Kluwer Academic Publishers, 1998.

[8] M. Nadler and Eric P. Smith, *Pattern Recognition Engeenering*, John Wiley&Sons Inc., 1993.

[9] P.Pudil, J. Navovicova, and J.Kittler, ‚Floating search methods in feature selection‘, *Pattern Recognition Letters,* **15**(1994), 1119-1125.

[10] J.R. Quinlan, C4.5:Programs for Machine Learning, Morgan Kaufmann, 1993.

[11] A.P. White and W.Z. Lui,‘Bias in the information-based measure in decision tree induction‘, *Machine Learning,* **15**(1994),321-329.