Exploiting Partial Decision Trees for Feature Subset Selection in e-Mail Categorization

Helmut Berger
iSpaces Group
Electronic Commerce
Competence Center – EC3
Donau-City-Straße 1
A–1220 Wien, Austria
helmut.berger@ec3.at

Dieter Merkl
Institut für Rechner-
gestützte Automation
Technische Universität Wien
Karlsplatz 13/183
A–1040 Wien, Austria
dieter@inso.tuwien.ac.at

Michael Dittenbach
iSpaces Group
Electronic Commerce
Competence Center – EC3
Donau-City-Straße 1
A–1220 Wien, Austria
michael.dittenbach@ec3.at

ABSTRACT

In this paper we propose PART$_{fs}$ which adopts a supervised machine learning algorithm, namely partial decision trees, as a method for feature subset selection. In particular, it is shown that an aggressive reduction of the feature space can be achieved with PART$_{fs}$ while still allowing for comparable classification results with conventional feature selection metrics. The approach is empirically verified by employing two different document representations and four different text classification algorithms that are applied to a document collection consisting of personal e-mail messages. The results show that a reduction of the feature space in the magnitude of ten is achievable without loss of classification accuracy.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval; H.3.1 [**Information Systems**]: Content Analysis and Indexing

General Terms

Indexing methods, information filtering, feature selection

Keywords

Text categorization, machine learning

1. INTRODUCTION

Feature subset selection aims at finding the smallest feature set having the most beneficial impact on machine learning algorithms, i.e. it's prime goal is to identify a subset of features upon which attention should be centered. Generally, the initial number of features extracted from arbitrary text corpora is very large. Most machine learning algorithms are computationally demanding and are not well suited for analyzing very high-dimensional feature spaces. If the number of features increases immoderately, some algorithms are neither able to perform their task in a reasonable amount of time nor with reasonable quality of results. To this end, feature subset selection strategies might be employed to reduce the search space while retaining those features that are potentially relevant to the learner. A wide range of studies corroborate that learning algorithms perform their classification task on a reduced subset of features with a marginal decrease in accuracy [13, 9]. Since feature subset selection is a common task in text categorization, it is of great importance in the context of e-Mail classification as well. Crawford et al. report in [6] that feature subset selection has positive impact on the classification performance in e-Mail categorization. Aggressive feature reduction to about 5% of the original number of features achieved yet feasible results. The findings described in [2] suggest that aggressive feature reduction in e-Mail categorization is especially advantageous when character n-gram document representation is used.

In this paper we introduce a new approach for feature subset selection, namely PART$_{fs}$. Basically, we outline the algorithm for deriving a set of features by exploiting the rule set generated by the decision tree learner PART [8]. The approach is empirically verified by employing two different document representations and four different text classification algorithms that are applied to a document collection consisting of personal e-mail messages. In particular, the following research questions are addressed by the experiments presented in this paper:

- Which magnitude of feature space reduction can be achieved with PART$_{fs}$?
- What influence has the document representation on the magnitude of feature space reduction?
- What is the overall performance of different classification approaches using the reduced feature sets?
- Will the new feature subset selection technique allow for comparable classification accuracies with conventional feature selection approaches?

This paper is structured as follows. Section 2 outlines partial decision trees and introduces the PART$_{fs}$ feature selection algorithm. A description of our experiments is provided in Section 3. Finally, Section 4 concludes with a discussion of our findings.

SAC'06 April 23-27, 2006, Dijon, France

2. FEATURE SELECTION WITH PARTIAL DECISION TREES

Generally, the initial number of features extracted from text corpora is very large. Due to the fact that most machine learning algorithms are computationally demanding, it is desirable to reduce the feature space while retaining those features that are potentially relevant. Feature selection strategies may be categorized into *wrapper*, *filter* and *embedded* approaches [10]. The distinguishing criterion is whether the method takes into account the characteristics of the data, the target concept or the learning algorithm. The goal in wrapper approaches is to find a subset of features that maximizes accuracy. This implies that the feature selection algorithm needs to derive an appropriate subset of features using the learning algorithm itself as an intrinsic element of the evaluation function. The same algorithm, however, is then applied to learn the final target concept. In filter approaches, the aim is to filter irrelevant or redundant features on the basis of the characteristics of the training data without involving any learning algorithm. Finally, in embedded approaches the feature selection process is done as part of the learning algorithm. A variety of feature selection metrics such as Information Gain, χ^2 , Principal Component Analysis have been applied in text classification and we refer to [7] for a recent survey. The proposed feature selection algorithm PART$_{fs}$ is a filter approach and is based on the decision tree learner PART [8].

2.1 Partial Decision Trees

Rule learners are prominent representatives of supervised machine learning approaches. Basically, this type of learner tries to induce a set of rules for a collection of training instances. These rules are then applied on the test instances for classification purposes. Two well-known members of the family of rule-learners are C4.5 [15] and RIPPER [5]. Both approaches perform two steps to induce their rule sets. First, an initial rule set is determined, and second, these rules are adjusted or discarded according to a global optimization strategy. C4.5, for instance, generates an unpruned decision tree and transforms this tree into a set of rules. For each path from the root node to a leaf a rule is generated. Then, each rule is simplified separately followed by a rule-ranking strategy. Finally, the algorithm deletes rules from the rule set as long as the rule set's error rate on the training instances decreases. RIPPER implements a divide-and-conquer strategy to rule induction. Only one rule is generated at a time and the instances from a training set covered by this rule are removed. It iteratively derives new rules for the remaining instances of the training set.

Frank and Witten describe a rule induction approach without the need for applying a global optimization strategy to generate appropriate rules [8]. PART (Partial Decision Trees) adopts the divide-and-conquer strategy of RIPPER and combines it with the decision tree approach of C4.5. More precisely, PART generates a set of rules according to the divide-and-conquer strategy, removes all instances from the training collection that are covered by this rule and proceeds recursively until no instance remains. To generate a single rule, PART builds a partial decision tree for the current set of instances and chooses the leaf with the largest coverage as the new rule. Afterwards, the partial decision tree is discarded which avoids early generalization.

Algorithm. PART$_{fs}$ feature selection

PART$_{fs}(I) \rightarrow RedF$
Input
I ... set of training instances
Output
$RedF$ ... a reduced subset of features

begin
$RedF \leftarrow \emptyset$
$Ruleset \leftarrow$ PART(I)
foreach *Rule* in *Ruleset* do
    $Features \leftarrow$ extractFeatures(*Rule*)
    $RedF \leftarrow RedF \cup Features$
done
return $RedF$
end

Figure 1: PART$_{fs}$ feature selection algorithm.

2.2 The PART$_{fs}$ Feature Selector

PART$_{fs}$ is a feature selection approach that further reduces the number of features already reduced with conventional feature selection metrics. More specifically, when applying PART$_{fs}$ a reduced subset of features is obtained. Figure 1 depicts the algorithm underlying PART$_{fs}$ that expects a set of training instances I as input and returns a (reduced) set $RedF$ of features. Firstly, the PART machine learning algorithm is applied on the training instances in order to build a set of decision rules (cf. Figure 1, $Ruleset$). Normally, these rules are used to perform the actual categorization task of test instances. However, this specific step is skipped in PART$_{fs}$ since we are just interested in those features that constitute each particular rule.

Figure 2 exemplifies a set of rules generated during PART's training phase. Each rule contains an arbitrary number of features associated by means of Boolean operators and a corresponding class. Note that binary feature weighting is used in this example and, thus, each listed feature is either explicitly present or absent in a particular instance. After PART's training phase terminates the algorithm steps through each rule of $Ruleset$ and extracts all features contained in the rule. Subsequently, the union of the newly extracted features ($Features$) and the set of reduced features ($RedF$) is calculated. Finally, the complete set of reduced features is derived; in this example $RedF$ ={complete, ec, froeschl, greiner, ifsegifstuwienacat, karlfroeschlecat, merkl, precedence, research, send, sender, textplain, xauthenticationwarning, xuid}.

```
...
(precedence = 0 AND karlfroeschlecat = 0
AND ifsegifstuwienacat = 1) → department

(froeschl = 0 AND greiner = 0 AND
xauthenticationwarning = 0 AND
sender = 0 AND textplain = 0) → misc

(ec = 0 AND precedence = 0 AND
xuid = 1 AND merkl = 1 AND
research = 0 AND send = 0 AND
complete = 0) → lectures
...
Number of Rules: 52
```

Figure 2: A sample of three decision rules.

Table 1: Corpus statistics.

Class	Messages		Words					Description
	per class	no content	Total	Mean	Min	Max	Standard Dev.	
admin	32	0	12,259	383.09	174	1,218	251.87	administration
dbworld	260	1	216,011	830.81	172	2,252	385.32	mailinglist
dilbert	70	0	78,221	1,117.44	866	2,951	334.27	"daily dilbert"
ec3	20	1	12,987	649.35	118	3,705	757.12	project related
department	30	1	8,592	286.4	96	691	182.99	department issues
isaus	24	2	19,909	829.54	288	2,443	616.5	mailinglist
kddnuggets	6	0	9,102	1,517.00	1,302	1,785	160.23	mailinglist
lectures	315	19	99,072	314.51	109	10,969	627.63	lecturing issues
michael	27	2	5,381	199.3	103	488	91.74	unspecific
misc	69	2	53,204	771.07	114	12,777	1,703.08	unspecific
paper	15	1	4,86	324	157	562	135.17	publications
position	66	0	34,033	515.65	259	895	131.31	job announcements
seworld	132	0	95,753	725.4	200	2,372	323.01	mailinglist
spam	701	90	297,423	424.28	72	5,234	508.02	spam messages
technews	31	0	44,258	1,427.68	1,045	1,571	95.01	mailinglist
talks	13	0	6,288	483.69	177	1,555	335.09	talk announcements
Total	**1,811**	**119**	**997,353**	**550.72**	**72**	**12,777**	**624.15**	

3. EMPIRICAL VALIDATION

The document collection consists of 1,811 e-mail messages. These messages have been collected during a period of four months commencing with October 2002 until January 2003. The e-mails have been received by a single personal e-mail account at the *Institut für Softwaretechnik*, Vienna University of Technology, Austria. At first, messages containing confidential information were removed from the corpus. Next, the corpus was manually categorized according to the classes outlined in Table 1. Note that the manual classification was performed a couple of months after the original collection which may have had some negative effect on the quality of the classification. Due to the manual classification of the corpus, some of the messages may have been misclassified. The classes might give the impression of a more or less artificial separation. However, introducing similar classes was intentionally done for assessing the performance of classifiers on closely related topics. Consider, for example, the *position* class which constitutes a set of 66 messages mainly posted via the *dbworld* and *seworld* mailinglists. In particular, it contains 38 *dbworld* messages, 23 *seworld* messages, 1 *isaus* message and 4 other messages. In contrast to standard *dbworld* or *seworld* messages, *position* messages deal with academic job announcements rather than academic conferences and alike.

3.1 Preprocessing

In order to determine the achievable magnitude of feature space reduction with respect to the document representation, two different types of document representations were employed. So, a *character n-gram* document representation [4] is compared against a *word based* document representation. In a nutshell, an n-gram is an n-character slice of a longer character string. When dealing with multiple words in a string, the blank character indicates word boundaries and is usually retained during the construction of the n-grams. However, it might be substituted with another special character. As an example for $n = 2$, the character *bi*-grams of *"topic spotting"* are {*to, op, pi, ic, c_, _s, sp, po, ot, tt, ti, in, ng*}. Note that the "space" character is part of the alphabet and represented by "_" in this example.

Both document representations comprise all data contained in the e-mail message, i.e. the complete header as well as the body. However, the e-mail header was not treated in a special way. All non-Latin characters, apart from the blank character, were discarded which entails that all HTML-tags remain part of the representation. We relied on binary weighting for both document representation approaches, i.e. just the presence or absence of an n-gram or word in the document is recorded. This decision was grounded upon [3] in which binary weighting resulted in superior categorization accuracy as compared to frequency-based weighting for this particular corpus. Note that no stemming was applied to the word-based document representation. Subsequently, all characters were translated to lower case. For each message of the set a character n-gram document representation with $n \in \{2, 3\}$ was generated and $20,413$ distinct features were obtained. In case of the word-based document representation $32,240$ features were counted. Next, we applied the χ^2 feature selection metric on both document representations and subsequently selected the m top-ranked features with $m \in \{2000, 1000, 500, 400, 300, 200, 100\}$. In a second step, we applied PART_{fs} on each χ^2-processed feature set in order to further reduce the feature space. Table 2 depicts the resulting feature sets. The first column denotes the feature sets generated with the χ^2 metric. Those features that are obtained after applying PART_{fs} on the word-based document representation are depicted in the second column. Interestingly, PART_{fs} reduces the 400 χ^2 feature set to 58 features while, in case of the 300 χ^2 feature set, the reduction was somewhat less effective (60 features). The third column gives the reduced feature sets for the n-gram document representation. Note that there are two entries for 189 features in case of PART_{fs}. However, these sets do not comprise the same features but, coincidentally, PART_{fs} reduced both, the 1,000 and 2,000 χ^2 feature sets, to 189 features.

3.2 Algorithms and Evaluation Measures

The major objective of the experiments is to determine the performance of different text classification approaches with respect to aggressive feature reduction in order to assess the

Table 2: Feature sets used in the experiments.

no. of χ^2 features	no. of PART_{fs} features	
word-based, n-grams	word-based	n-grams
100	9	73
200	29	140
300	60	162
400	58	165
500	67	167
1,000	126	189
2,000	149	189

Table 3: Macro-averaged F-Measure values.

χ^2	SMO	IBk	NBm	PART
100	**0.420**	0.417	0.373	0.419
200	**0.603**	0.571	0.477	0.592
300	**0.779**	0.703	0.598	0.750
400	**0.796**	0.698	0.590	0.770
500	**0.820**	0.732	0.670	0.795
1,000	**0.841**	0.767	0.821	0.790
2,000	0.850	0.759	**0.856**	0.786

(a) Word-based using χ^2 features.

PART_{fs}	SMO	IBk	NBm	PART
9	**0.419**	0.418	0.214	**0.419**
29	**0.611**	0.577	0.486	0.587
60	**0.786**	0.699	0.676	0.754
58	**0.803**	0.691	0.725	0.768
67	**0.821**	0.735	0.782	0.780
126	**0.824**	0.767	0.787	0.808
149	**0.847**	0.741	0.807	0.790

(b) Word-based using PART_{fs} features.

χ^2	SMO	IBk	NBm	PART
100	**0.733**	0.631	0.595	0.650
200	**0.778**	0.683	0.675	0.697
300	**0.801**	0.737	0.730	0.704
400	**0.821**	0.752	0.751	0.715
500	**0.833**	0.756	0.756	0.752
1,000	**0.849**	0.776	0.769	0.727
2,000	**0.846**	0.771	0.781	0.729

(c) n-grams using χ^2 features.

PART_{fs}	SMO	IBk	NBm	PART
73	**0.730**	0.615	0.602	0.672
140	**0.778**	0.678	0.685	0.677
162	**0.799**	0.731	0.753	0.716
165	**0.808**	0.735	0.763	0.713
167	**0.821**	0.716	0.769	0.734
189	**0.840**	0.745	0.766	0.750
189	**0.831**	0.733	0.783	0.745

(d) n-grams using PART_{fs} features.

effectiveness of PART_{fs} in e-Mail categorization. Note that all experiments were carried out with *10-fold cross validation* to avoid effects that can be attributed to a particular split into training and test sets [12]. Representatives of four different machine learning philosophies were selected for our experiments. All applied classifiers are supervised learning approaches. In particular, classifiers of the following machine learning areas were chosen: (i) a Naïve Bayes classification approach as a representative of Bayesian learning [11], (ii) IBk as a representative of instance-based learning [1], (iii) Support vector machines trained with the sequential minimal optimization algorithm as a representative of kernel-based learning [14] and, (iv) PART as a representative of decision tree learning [8].

The effectiveness of text classification algorithms is evaluated by means of the F–measure as described in [17]. It combines the standard precision P (defined as the ratio of relevant documents and total number of documents in the collection), and recall R (defined as the ratio of relevant documents retrieved and the total number of relevant documents in the collection) with an equal weight as $F(P,R) = \frac{2 \cdot P \cdot R}{P+R}$. In particular, we used the macro-averaged F–measure that calculates an F–measure value for each individual category which is averaged over the results of the different categories.

The percentage of correctly classified instances is assessed by the *Accuracy* measure. It calculates the proportion of the number of correctly classified instances on the total number of instances in the collection.

3.3 Experimental Results

Table 3 gives a comparison of the macro-averaged F–measure values calculated for each classifier. Each classification algorithm was applied on the χ^2 feature set as well as on the PART_{fs} reduced feature set in the context of the two different document representations, namely the character n-gram representation and the word-based representation. Note that *SMO* in the headings of Table 3 refers to the support vector machine using the sequential minimal optimization training algorithm. *IBk* identifies the instance-based learner with $k = 5$ and the multi-nomial Naïve Bayes classifier is referred to as *NBm*. *PART* refers to the partial decision tree classifier. We used the implementation of the learning algorithms as provided with the WEKA toolkit [16].

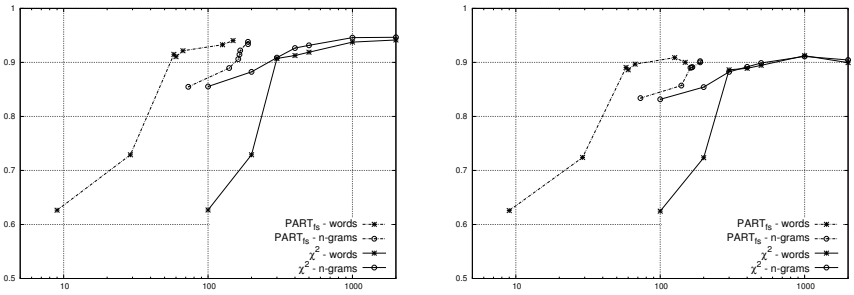
In case of the word-based document representation (cf. Table 3(a) and (b)) the F–measure values show a fairly similar trend for both feature sets. More precisely, the classifiers achieved the same or even better F–measure values using the PART_{fs} reduced sets, which comprise far less features than the χ^2 feature sets. Rather amazing are the results obtained with 9 PART_{fs} features which show almost the same F–measure values as those achieved with 100 features in case of the χ^2 feature set. Similar results are obtained with the character n-grams, cf. Table 3(c) and (d).

In Figure 3 the classification accuracy (y–axis) of the text classifiers for the various numbers of features (x–axis) is given. Each curve corresponds to a "feature selection strategy – document representation" tuple. Note that the x–axis is logarithmically scaled. When considering the support vector machine, cf. Figure 3(a), the accuracy follows a similar trend for both feature selection strategies although the magnitude of features used in case of PART_{fs} is about ten times smaller. This can be observed for both document representations. The accuracies achieved by the other classifiers develop in a rather similar manner.

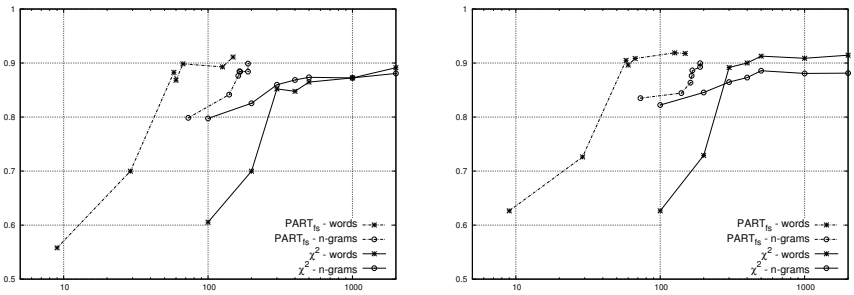
4. CONCLUSION

In this paper, the feature subset selection approach PART_{fs} was introduced. More precisely, PART_{fs} exploits the partial decision tree learning algorithm for feature space reduction in a multi-class e-mail categorization setting. The corpus consists of personal e-mail messages which were manually split into multiple classes. To verify the ability of PART_{fs} to further reduce the feature space, two different document representations and four text classification algorithms were chosen to empirically assess the classification performance.

When looking at the results from the experiments as presented above an immediate observation is that the feature space can be reduced by the magnitude of 10 while achieving similar classification results. For example, it takes about 2,000 χ^2 features to achieve similar accuracies as those obtained with 149 PART_{fs} features. In general, all four classifiers performed their classification task with comparable performance when applied to the PART_{fs} feature sets. This observation holds true for both the word based document representation and the character n-gram document representation. So, the document representation has little influence on the relevance of features selected by PART_{fs}. How-



(a) Support Vector Machine (SMO) (b) Instance Based Learner (IBk)



(c) Multinomial Naive Bayes (NBm) (d) Partial Decision Tree (PART)

Figure 3: Classification accuracy of individual classifiers using χ^2 and PART$_{fs}$ reduced feature sets.

ever, it turned out that the magnitude of achievable feature reduction in case of the n-gram document representation is smaller than with the word based approach. In this case for example, a reduction from 100 χ^2-ranked features to 9 features can be obtained with the PART$_{fs}$ feature selection approach. Contrary, the 100 χ^2-ranked features in the n-gram document representation were reduced to 73 PART$_{fs}$ features, cf. Table 2. It is interesting to note that the degradation of classification accuracy is far less dramatic in case of the n-gram document representation. That applies to both the χ^2-ranked feature set and those sets comprising features selected with PART$_{fs}$. Similar observations regarding the effects of the n-gram document representation in case of aggressive feature space reduction are reported in [3].

5. REFERENCES

[1] D. Aha, D. Kibler, and M. Albert. Instance-Based Learning Algorithms. *Machine Learning*, 6(1), 1991.

[2] H. Berger, M. Köhle, and D. Merkl. On the Impact of Document Representation on Classifier Performance in eMail Categorization. In *Proc. Int'l Conf. Information Systems Technology and its Applications*, 2005.

[3] H. Berger and D. Merkl. A Comparison of Text-Categorization Methods applied to N-Gram Frequency Statistics. In *Proc. of the 17th Australian Joint Conf. on Artificial Intelligence*, 2004.

[4] W. B. Cavnar and J. M. Trenkle. N-gram-based text categorization. In *Proc. Int'l Symp. on Document Analysis and Information Retrieval*, 1994.

[5] W. W. Cohen. Fast effective rule induction. In *Proc. of the Int'l Conf. on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.

[6] E. Crawford, I. Koprinska, and J. Patrick. Phrases and feature selection in e-mail classification. In *Proc. 9th Australasian Document Computing Symp.*, 2004.

[7] G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, 2003.

[8] E. Frank and I. H. Witten. Generating accurate rule sets without global optimization. In *Proc. of the Int'l Conf. on Machine Learning*, pages 144–151. Morgan Kaufmann Publishers Inc., 1998.

[9] M. A. Hall and G. Holmes. Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1437–1447, 2003.

[10] G. H. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In *Proc. of the 11th Int'l Conf. on Machine Learning*, pages 121–129, 1994.

[11] A. McCallum and K. Nigam. A Comparison of Event Models for Naive Bayes Text Classification. In *Proc. of AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[12] T. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[13] D. Mladenic. Feature subset selection in text-learning. In *Proc. of the 10th European Conf. on Machine Learning*, pages 95–100, UK, 1998.

[14] J. Platt. Fast Training of Support Vector Machines using Sequential Minimal Optimization. In *Advances in Kernel Methods - Support Vector Learning*, pages 185–208. MIT Press, 1999.

[15] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., 1993.

[16] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, San Francisco, 2000.

[17] Y. Yang and X. Liu. A re-examination of text categorization methods. In *Proc. of the Int'l ACM SIGIR Conf. on R&D in Information Retrieval*, 1999.