Text Categorization with Many Redundant Features: Using Aggressive Feature Selection to Make SVMs Competitive with C4.5

Evgeniy Gabrilovich GABR@CS.TECHNION.AC.IL
Shaul Markovitch SHAULM@CS.TECHNION.AC.IL
Computer Science Department, Technion—Israel Institute of Technology, 32000 Haifa, Israel

Abstract

Text categorization algorithms usually represent documents as bags of words and consequently have to deal with huge numbers of features. Most previous studies found that the majority of these features are relevant for classification, and that the performance of text categorization with support vector machines peaks when no feature selection is performed. We describe a class of text categorization problems that are characterized with many *redundant* features. Even though most of these features are relevant, the underlying concepts can be concisely captured using only a few features, while keeping all of them has substantially detrimental effect on categorization accuracy. We develop a novel measure that captures feature redundancy, and use it to analyze a large collection of datasets. We show that for problems plagued with numerous redundant features the performance of C4.5 is significantly superior to that of SVM, while aggressive feature selection allows SVM to beat C4.5 by a narrow margin.

1. Introduction

Text categorization deals with assigning category labels to natural language documents. Categories come from a fixed set of labels, and each document may be assigned one or more categories. The absolute majority of works in the field employ the so-called "bag of words" approach and use plain language words as features (Sebastiani, 2002). Using a bag of words usually leads to an explosion in the number of features, so that even moderately-sized test collections often have thousands or even tens of thousands of features. In such high-dimensional spaces, feature selection (FS) is often necessary to reduce noise and avoid overfitting. Prior studies found support vector machines (SVM) and K-Nearest Neighbor (KNN) to be the best performing algorithms for text categorization (Dumais et al., 1998; Yang & Liu, 1999).

Joachims (1998) found that *support vector machines* are very robust even in the presence of numerous features, and further observed that the multitude of text features are indeed useful for text categorization. To substantiate this claim, Joachims used a Naive Bayes classifier with feature sets of increasing size, where features were first ordered by their discriminative capacity (using the information gain criterion), and then the most informative features were *removed*. The classifier trained on the remaining "low-utility" features performed markedly better than random labeling of documents with categories, thus implying that all features are relevant and should be used. These findings were later corroborated in more recent studies (Brank et al., 2002; Rogati & Yang, 2002) that observed either no improvement or even small degradation of SVM performance after feature selection. On the 20 Newsgroups collection (Lang, 1995), which is one of the standard text categorization datasets, feature selection significantly degrades the accuracy of SVM classification (Bekkerman, 2003) due to a very large and diversified vocabulary of newsgroup postings. Consequently, many later works using SVMs did not perform any feature selection at all (Leopold & Kindermann, 2002; Lewis et al., 2004).

In this paper we describe a class of text categorization problems that are characterized by many *redundant* features. The corresponding datasets were collected in the course of our prior work (Davidov et al., 2004), where we proposed a methodology for parameterized generation of labeled datasets for text categorization based on the Open Directory Project (ODP). In our present work we use a subset of 100 datasets whose categorization difficulty (as measured by baseline SVM accuracy) is evenly distributed from very easy to very hard. We observed that even though the datasets dif-

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004.

fer significantly in their difficulty, many of them are comprised of categories that can be told apart using a small number of words. For example, consider distinguishing the documents about Boulder, Colorado, from those about Dallas, Texas. A few proper names of local landmarks and a handful of words describing local industries and other peculiarities often suffice to distinguish texts about the two cities. Given these discriminators, other words add little differentiation power, and are therefore *redundant*. As we show in Section 3, support vector machines—which are usually quite robust in the presence of many features—do not fare well when a few good discriminators are vastly outnumbered by features with little *additional differentiation power*.

We further demonstrate that on such datasets C4.5 significantly outperforms SVM and KNN, although the latter are usually considered substantially superior text classifiers. When no feature selection is performed, C4.5 constructs small decision trees that capture the concept much better then either SVM or KNN. Surprisingly, even when feature selection is optimized for each classifier, C4.5 formulates a powerful classification model, significantly superior to that of KNN and only marginally less capable than that of SVM. We also show the crucial importance of aggressive feature selection for this class of problems on a different document representation. In this experiment we extend the conventional bag of words with features constructed using the WordNet electronic dictionary by generalizing original words; again, SVM performance steadily increases as fewer features are selected.

To account for this phenomenon, we developed a novel measure that predicts feature redundancy in datasets. This measure analyzes the distribution of features by their information gain, and reliably predicts whether feature selection will be beneficial or harmful for a given dataset. Notably, computation of this measure does not require to actually build a classifier, nor to invoke it on a validation set to determine an optimal feature selection level.

The main contributions of this paper are threefold. First, we describe a class of text categorization problems that have many redundant features, and for which aggressive feature selection is essential to achieve decent level of SVM performance. The existence of such class of problems is in contrast to most of prior research in text categorization, which found the majority of features (except the rarest ones) to be relevant, and specifically beneficial for SVM classification. Second, we use two different feature sets to show that without an aggressive feature selection, SVM classification is substantially inferior to that of C4.5, which was previously shown to be a less capable text classifier. Finally, we develop a measure that, given a dataset, predicts whether feature selection would be beneficial for it. This measure performs outlier detection in the distribution of features by information gain, without actually classifying the documents.

2. Experimental methodology

We conducted a series of experiments to explore the utility of feature selection for datasets plagued with redundant features. In what follows, we first describe the construction of the datasets used in the experiments, and then proceed to developing a measure that predicts the utility of feature selection for a given dataset.

2.1. Datasets

Acquiring datasets for text categorization based on Web directories has been often performed in prior studies, which used Yahoo! (Mladenic & Grobelnik, 1998), ODP (Chakrabarti et al., 2002; Cohen et al., 2002) and the Hoover's Online company database (Yang et al., 2002). This approach allows to eliminate the huge manual effort required to actually label the documents, by first selecting a number of categories (= directory nodes) to define the labels, and then collecting the documents from the subtrees rooted at these categories to populate the dataset.

In our prior work (Davidov et al., 2004) we developed a methodology for automatically acquiring labeled datasets for text categorization from hierarchical directories of documents, and implemented a system that performed such acquisition based on the Open Directory Project (`http://dmoz.org`). In the present paper we use a subset of 100 datasets acquired using this methodology. Each dataset consists of a pair of ODP categories with an average of 150 documents, and corresponds to a binary classification task of telling these two categories apart (documents are single-labeled, that is, every document belongs to exactly one category). When generating datasets from Web directories, where each category contains links to actual Internet sites, we construct text documents representative of those sites. Following the scheme introduced by Yang et al. (2002), each link cataloged in the ODP is used to obtain a small representative sample of the target Web site. To this end, we crawl the target site in BFS order, starting from the URL listed in the directory. A predefined number of Web pages (5 in this work) are downloaded, and concatenated into a *synthetic document*, which is then filtered to remove HTML markup; the average document size

after filtering is 11.2 Kilobytes.

The datasets vary significantly by their difficulty for text categorization, and baseline SVM accuracy obtained on them is nearly uniformly distributed between 0.6 and 0.92. To list a few examples, datasets in our collection range from easy ones containing such pairs of ODP categories as `Games/Video_Games/Shooter` and `Recreation/Autos/Makes_and_Models/Volkswagen`, to medium difficulty ones with `Arts/Music/Bands_and_Artists` VS. `Arts/Celebrities`, to hard ones such as `Regional/North_America/United_States/Virginia/Richmond/Business_and_Economy` VS. `Regional/North_America/United_States/Florida/Fort_Myers/Business_and_Economy`. The full collection of 100 datasets, along with additional statistics and all the raw data used in our experiments is available at `http://techtc.cs.technion.ac.il/techtc100` .

2.2. Predicting the utility of feature selection

In Section 3 we show that the majority of datasets we used in this study benefit greatly from aggressive feature selection. We conjectured that these datasets have a small number of features that together allow to learn the underlying concept concisely, while the rest of the features do more harm than good. To understand this phenomenon, we examined the distribution of features in each dataset by their information gain.

Figure 1 shows this distribution for several sample datasets.[1] Empirically, we observed that datasets with feature distribution similar to Dataset 46 benefit from feature selection immensely (for this particular dataset, aggressive feature selection improved SVM accuracy from 0.60 to 0.93). Such datasets have several features with high information gain, while the rest of their features have markedly lower IG scores. In contrast to these, datasets similar to Dataset 1 are characterized with smooth spectrum of IG values—in such cases feature selection will often eliminate features that carry essential information; indeed, for this dataset feature selection caused SVM accuracy to drop from 0.86 to 0.74. For comparison, we show a similarly looking graph for the 20 Newsgroups (20NG) dataset, which is often used for text categorization experiments and for which feature selection was found particularly harmful (Bekkerman, 2003).

Interestingly, high IG values of best-scoring features do not necessarily imply that feature selection will substantially improve the accuracy. For instance, Dataset 31 has several features with very high information gain, but its IG graph declines gracefully over

[1]Dataset ids refer to the full listing table at `http://techtc.cs.technion.ac.il/techtc100` .

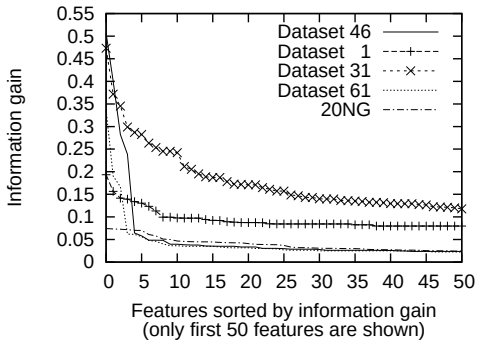

Figure 1. Distribution of features by IG in several datasets.

subsequent features, and does not fall as sharp as for Dataset 46. Consequently, feature selection only improves SVM accuracy from 0.92 to 0.95—a much more modest gain than for Dataset 46. On the other hand, Dataset 61 has somewhat lower initial IG values, but its IG graph declines very sharply. Feature selection was shown to be of high utility for this dataset as well, boosting the accuracy from 0.64 to 0.84.

The above results imply that the absolute values of information gain are of less importance than the *speed of decline* of IG values across features. To quantify this phenomenon, we need to assess the number of *outliers*—features whose information gain is highly above that of all other features. Under this definition the desired measure becomes easy to formulate. We first compute the information gain for all features, and then count the number of features whose information gain is higher than 3 standard deviations above the average. Although the underlying distribution cannot be assumed to be normal, this familiar statistical criterion works very reliably in practice. Formally, let $\mathcal{D}$ be a dataset and let $\mathcal{F}$ be a set of its features. We define the *Outlier Count (OC)* as

$$OC(\mathcal{D},\mathcal{F}) = |\{f \in \mathcal{F} : IG(f) > \mu_{IG} + 3 \cdot \sigma_{IG}\}| ,$$

where μ_{IG} and σ_{IG} are the average and standard deviation of information gain of the features in $\mathcal{F}$. In Section 3 we show that Outlier Count reliably predicts the utility of feature selection for a variety of datasets.

2.3. Extended feature set based on WordNet

Several studies in text categorization performed feature construction using the WordNet electronic dictionary (Fellbaum, 1998). In this work we show that aggressive feature selection can significantly improve categorization accuracy for document representation extended with constructed features.

Scott and Matwin (1999), and later Wermter and Hung (2002), used WordNet to change document representation from a bag of words to a bag of synsets (WordNet

notion of concepts), by using the hypernymy relation to generalize word senses. Since many words are not found in WordNet (e.g., neologisms, narrow technical terms, and proper names), we opted for *extending* a bag of words with WordNet-based features rather than completely changing document representation to a bag of synsets. To this end, we first perform feature generation by generalizing document words using WordNet, and then decimate the generated features through feature selection. In Section 3.4 we demonstrate that feature selection is as important for generated features as it is for regular features (plain language words).

2.4. Feature selection algorithms

A variety of feature selection techniques have been tested for text categorization, while Information Gain, χ^2, Document Frequency (Yang & Pedersen, 1997; Rogati & Yang, 2002), Bi-Normal Separation (Forman, 2003) and Odds Ratio (Mladenic, 1998) were reported to be the most effective. Adopting the probabilistic notation from Sebastiani (2002), we use $P(t_k, c_i)$ to denote the joint probability that a random document contains term t_k and belongs to category c_i, and N to denote the number of training documents. The above feature selection techniques are then defined as follows:

1. Information Gain (IG): $\sum_{c\in\{c_i,\overline{c_i}\}}\sum_{t\in\{t_k,\overline{t_k}\}} P(t,c)\cdot\log\frac{P(t,c)}{P(t)P(c)}$
2. χ^2 (CHI): $N\cdot\frac{P(t_k,c_i)P(\overline{t_k},\overline{c_i})-P(t_k,\overline{c_i})P(\overline{t_k},c_i)}{P(t_k)P(\overline{t_k})P(c_i)P(\overline{c_i})}$
3. Document Frequency (DF): $N\cdot P(t_k)$
4. Bi-Normal Separation (BNS): $|F^{-1}(P(t_k|c_i)) - F^{-1}(P(t_k|\overline{c_i}))|$, where F is the cumulative probability function of the standard Normal distribution
5. Odds Ratio (OR): $\frac{P(t_k|c_i)\cdot(1-P(t_k|\overline{c_i}))}{(1-P(t_k|c_i))\cdot P(t_k|\overline{c_i})}$
6. Random (RND)

Actual feature selection is performed by selecting the top scoring features, using either a predefined threshold on the feature score or a fixed percentage of all the features available. In addition to these "principled" selection schemes, we unconditionally remove stop words and words occurring in less than three documents.

2.5. Classification algorithms and measures

We used the datasets described in Section 2.1 to compare the performance of *Support Vector Machines* (Vapnik, 1995), *C4.5* (Quinlan, 1993), and *K-Nearest Neighbor* (Duda & Hart, 1973). In this work we used the SVM^{light} implementation (Joachims, 1999) with a linear[2] kernel.

We used classification accuracy as a measure of text categorization performance. Studies in text categorization usually work with multi-labeled datasets in which each category has much fewer positive examples than negative ones. In order to adequately reflect categorization performance in such cases, other measures of performance are conventionally used (Sebastiani, 2002), including precision, recall, F_1, and precision-recall break-even point (BEP). However, for single-labeled datasets all these measures can be proved to be equal to accuracy, which is the measure of choice in the machine learning community.

3. Empirical evaluation

In this section we evaluate the role of feature selection for several classification algorithms operating on datasets with many redundant features. We conducted the experiments using a text categorization platform of our own design and development called $\mathcal{H}$OGWARTS. All accuracy values reported below were obtained using 4-fold cross-validation scheme.

When working with support vector machines, it is essential to perform adequate parameter tuning. In the case of a linear kernel (and under the assumption of equal cost of errors on positive and negative examples), the only relevant parameter is C, namely, the trade-off between training error and margin. To optimize this parameter, we set aside one fold of the training data as a validation set, and for each feature selection level selected the best C value from among $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3, 10^4\}$.

3.1. Validation of $\mathcal{H}$OGWARTS performance

In this section we demonstrate that the results of classifying existing datasets with $\mathcal{H}$OGWARTS are consistent with those in other published studies. Figure 2 shows the results of using SVM in conjunction with IG feature selection to classify three datasets frequently used in text categorization studies: 10 largest categories of Reuters-21578 (Reuters, 1997), 20 Newsgroups (Lang, 1995), and Movie Reviews (Pang et al., 2002).[3] Using all features, $\mathcal{H}$OGWARTS achieved BEP

[2] Joachims (1998) observed that most text categorization problems are linearly separable, and consequently most studies in the field used a linear SVM kernel (Bekkerman, 2003; Forman, 2003; Brank et al., 2002).

[3] Since the former two of these datasets are multi-labeled, we use precision-recall break-even point (BEP) as a measure of classification performance rather than accuracy (see Section 2.5).

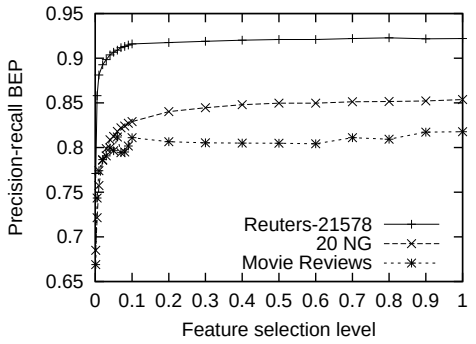

Figure 2. HOGWARTS performance on existing datasets (feature selection with IG).

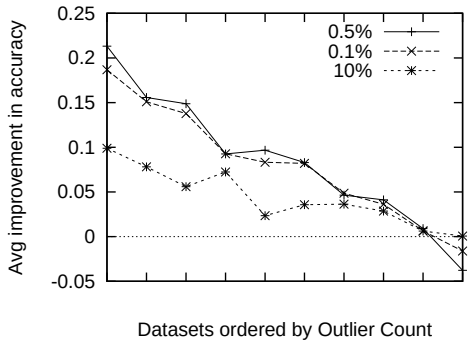

Figure 3. Improvement in SVM accuracy at different FS levels vs. using 100% features.

of 0.922 on Reuters, 0.854 on 20 Newsgroups and 0.818 on Movie Reviews. These results are very similar to the performance obtained by other researchers (all using SVM). Dumais et al. (1998) achieved BEP of 0.92 for the 10 largest Reuters categories. Bekkerman (2003) obtained BEP of 0.856 on the 20 Newsgroups dataset. Pang et al. (2002) obtained accuracy of 0.829 on the Movie Reviews dataset.

As can be seen in Figure 2, any level of feature selection harms the performance on all of these datasets. The graphs for χ^2 and BNS feature selection algorithms exhibit behavior very similar to IG, so we do not show them here owing to lack of space. Note that all the experiments reported in the rest of the paper use the 100 datasets we acquired as explained in Section 2.1.

3.2. Predicting the utility of feature selection with Outlier Count

We now show that the Outlier Count measure defined in Section 2.2 reliably predicts the utility of feature selection. Figure 3 shows the improvement in SVM accuracy at several feature selection levels versus the baseline accuracy obtained using 100% of features. As we can see, Outlier Count strongly correlates with the magnitude of improvement that can be obtained through feature selection. We observe that at lower values of Outlier Count aggressive feature selection is highly beneficial. Conversely, at higher OC values much more moderate (if any) feature selection should be performed, while aggressive selection causes degradation in accuracy. The next section examines the correlation of Outlier Count with the differences in performance between individual classifiers.

The Outlier Count for the datasets we used is nearly uniformly distributed between 6 and 62, with a single outlier value (no pun intended!) of 112 for Dataset 1 (Figure 1), for which feature selection caused SVM accuracy to drop from 0.86 to 0.74. For other datasets frequently used for text categorization, Outlier Count for Reuters-21578 is 78, Movie Reviews—154, and 20 Newsgroups—391, which explains why feature selection does for them more harm than good.

Based on these findings, we conclude that using Outlier Count for ordering datasets reflects the degree to which a dataset can be concisely described by only a few features, while the rest of the features are predominantly redundant and have detrimental effect on classification results.

3.3. Comparison of classifiers

Figure 4 compares the performance of SVM, KNN and C4.5 on the 100 datasets ordered by Outlier Count. When no feature selection is employed, the performance of C4.5 mostly dominates that of SVM and KNN, and only declines in the rightmost part of the graph, which contains datasets where a few features are not sufficient for learning the concept.

Table 1 shows classifier accuracy without feature selection and with the optimal feature selection level for each classifier. We used *paired t-test* to assess the significance of differences in classifier accuracy over the 100 datasets (see Table 2). Without any feature selection, the differences between classifiers were found to be very significant at $p < 5 \cdot 10^{-3}$ or lower. For individual classifiers, the improvement in accuracy due to feature selection was extremely significant at $p < 10^{-13}$.

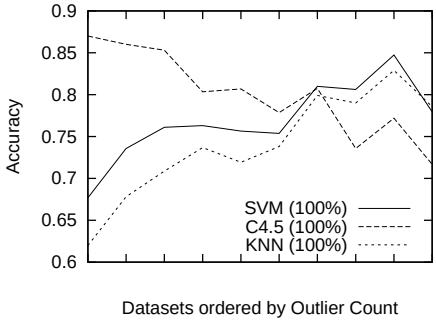

Figure 4. Comparison of performance of SVM, C4.5 and KNN with 100% features.

Table 1. Classifier accuracy at different FS levels.

Classifier	Accuracy with 100% features	Accuracy with the optimal FS level
SVM	0.769	0.853 (using 0.5% features)
C4.5	0.800	0.843 (using 0.5% features)
KNN	0.741	0.827 (using 2% features)

Table 2. Statistical significance of differences in classifier accuracy (p values).

Classifier (FS level)	C4.5 (100%)	KNN (100%)	SVM (0.5%)	C4.5 (0.5%)	KNN (2%)
SVM (100%)	$5 \cdot 10^{-3}$	$4 \cdot 10^{-9}$	$4 \cdot 10^{-15}$	$2 \cdot 10^{-10}$	$6 \cdot 10^{-11}$
C4.5 (100%)		$2 \cdot 10^{-5}$	$6 \cdot 10^{-14}$	$2 \cdot 10^{-15}$	$3 \cdot 10^{-4}$
KNN (100%)			$2 \cdot 10^{-16}$	$6 \cdot 10^{-13}$	$6 \cdot 10^{-14}$
SVM (0.5%)				$9 \cdot 10^{-3}$	$4 \cdot 10^{-8}$
C4.5 (0.5%)					$5 \cdot 10^{-3}$

3.4. The effect of using different feature sets

Figure 5 compares the performance of classifiers at different feature selection levels (using Information Gain). As we can see, C4.5 performs better than SVM except for the most aggressive FS levels, where their accuracy becomes nearly equal. Interestingly, C4.5 stays high above KNN at most FS levels.

Figure 6 presents a similar graph for the extended feature set based on WordNet. Here we use all features of the conventional bag of words, and only apply feature selection to the constructed features. C4.5 clearly manages the multitude of redundant features much better than both SVM and KNN. It is also noteworthy that the accuracy of SVM and KNN increases steadily as feature selection becomes more aggressive, while the improvement in their performance with 0.5% features vs. 100% features is strongly significant at $p < 10^{-18}$.

When using the optimal FS level (0.5% for both regular words and WordNet concepts), the addition of WordNet features is only responsible for a minor improvement in SVM accuracy from 0.853 to 0.854.

3.5. The effect of using different FS algorithms

Figures 7 and 8 show the effect of using different feature selection algorithms (see Section 2.4) with SVM and C4.5. Consistently with prior studies (Forman, 2003; Rogati & Yang, 2002), we observe that IG, CHI and BNS are the best performers, while the difference between them is not statistically significant.[4] In contrast with prior studies, we observe that on the family of datasets we described, the best performance of SVM is obtained when only using a tiny fraction of features (0.5% for the three best FS techniques).

[4]The graph for KNN looks substantially similar and also confirms the superiority of IG, CHI and BNS (with negligible differences), so we omit it owing to lack of space.

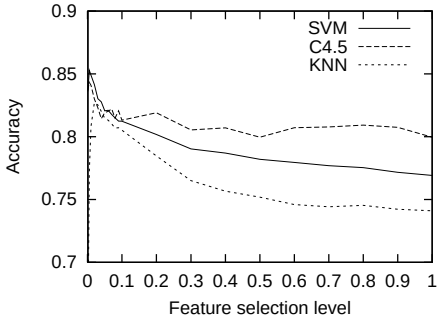

Figure 5. Classification using a bag of words.

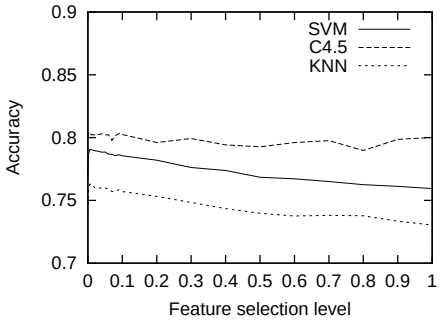

Figure 6. Classification using an extended feature set.

3.6. Testing the relevancy of features

In previous sections we showed that text categorization can greatly benefit from aggressive feature selection. We now address the question whether the features discarded by selection are at all relevant for classification. Following Joachims (1998), we sorted all features by their information gain, and then removed progressively larger fractions (0.1%, 0.5%, 1%, ..., 10%, 20%, ..., 100%) of the *most informative* features. As can be seen in Figure 9, the performance of an SVM classifier trained on the remaining features is noticeably better than random up to very high levels of such harmful "selection". These results corroborate earlier findings by Joachims (1998), and support our hypothesis that the features removed through selection are *redundant*, even though most of them may be considered relevant.

4. Discussion

Studies in text categorization usually represent documents as a bag of words, and consequently have to manage feature spaces of very high dimensionality. Most previous works in the field found that these numerous features are relevant for classification, and that in particular the performance of SVM text categorization peaks when no feature selection is performed.

We described a class of datasets plagued with *redundant* features, such that their elimination significantly boosts categorization accuracy of a host of classifiers. Specifically, we showed that when no feature selection

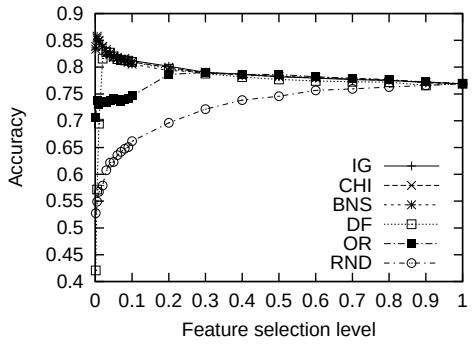

Figure 7. SVM accuracy vs. FS level.

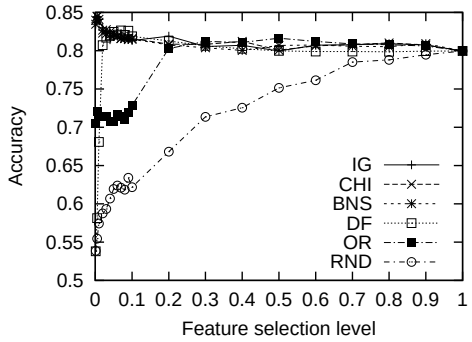

Figure 8. C4.5 accuracy vs. FS level

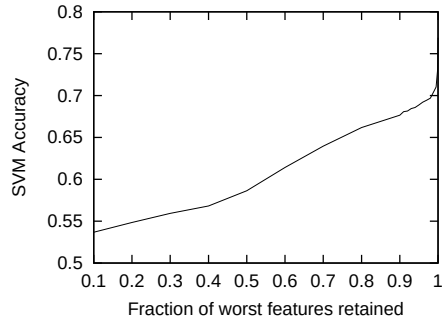

Figure 9. Removing the best features by IG.

is employed on such datasets, SVMs are significantly outperformed by C4.5. To explain this phenomenon, we analyzed the distribution of features by their information gain, and observed that this effect occurs when a small number of features are sufficient for concisely learning the underlying concept. We defined a measure named Outlier Count that, for a given dataset and fixed representation scheme, estimates the amount of feature redundancy through outlier analysis.

In a series of experiments, we demonstrated that Outlier Count reliably predicts the amount of improvement that can be gained through feature selection. These findings are backed by empirical evidence both for the conventional bag of words, and for a representation extended through feature generation based on WordNet. We further performed a controlled ablation study to verify that the redundant features are in fact relevant for classification. To this end, we removed progressively larger fractions of most informative features, and found the remaining ones to suffice for better than random performance. Finally, we analyzed several established benchmarks for text categorization with respect to Outlier Count, and explained why they do not benefit from feature selection.

Following the established practice in text categorization, throughout this paper we used an SVM classifier with a linear kernel. In an ancillary experiment we sought to determine whether a non-linear SVM kernel may fare better than a linear one when dealing with numerous redundant features. Without feature selection, switching from a linear kernel to an RBF one reduced the accuracy from 0.769 to 0.687. Even at the optimal feature selection level, the accuracy achieved with an RBF kernel was slightly below that of a linear one (0.849 vs. 0.853), contradicting our anticipation of better performance by a more sophisticated kernel. However, this experiment should be considered preliminary, and in our future work we plan to conduct a thorough investigation of the ability of non-linear SVM kernels to withstand high rates of redundant features.

In a recent study, Forman (2003) proposed a novel feature selection algorithm named Bi-Normal Separation, which improved the performance of SVM text categorization on a range of datasets. Peak performance was obtained when using 500–1000 features (approximately 10% of all available features on the average). More aggressive feature selection led to sharp degradation of the results—using less than 100 features caused macro-F_1 to decrease by 5%–10% depending on the selection algorithm used.

Our work corroborates the findings that feature selection can help text categorization with SVMs, and describes a class of problems where the improvement due to feature selection is particularly large. We showed that for this class of problems the improvement in accuracy can be twice as high as found by Forman (2003) (namely, 8.4% vs. 4.2%), while optimal performance is achieved when using much fewer features (between 5 and 40, depending on the dataset). We also evaluated several feature selection algorithms on text categorization problems characterized with many redundant features. Our results support earlier findings that Information Gain, Bi-Normal Separation and χ^2 are the most powerful feature selection algorithms, while the differences between them are not significant.

It should be noted that for all the datasets we used, the utility of feature selection could be established by setting aside part of the training data to serve as a validation set. Indeed, the high redundancy level was

so pronounced, that the optimal selection level for the testing data could almost always be correctly determined on the validation fold. However, we believe that the introduction of Outlier Count and the use of ablation experiments that systematically eliminate most informative features, allow deeper understanding of the issues of feature redundancy and relevancy.

Acknowledgments

We thank Ran El-Yaniv for advice on SVM tuning.

References

Bekkerman, R. (2003). Distributional clustering of words for text categorization. Master's thesis, CS Department, Technion—Israel Inst. of Technology.

Brank, J., Grobelnik, M., Milic-Frayling, N., & Mladenic, D. (2002). Interaction of feature selection methods and linear classification models. *Workshop on Text Learning held at ICML-2002*.

Chakrabarti, S., Joshi, M. M., Punera, K., & Pennock, D. M. (2002). The structure of broad topics on the web. *Proc. of the Int'l World Wide Web Conference*.

Cohen, D., Herscovici, M., Petruschka, Y., Maarek, Y. S., Soffer, A., & Newbold, D. (2002). Personalized pocket directories for mobile devices. *Proc. of the Int'l World Wide Web Conference*.

Davidov, D., Gabrilovich, E., & Markovitch, S. (2004). Parameterized generation of labeled datasets for text categorization based on a hierarchical directory. *To appear in SIGIR'04*.

Duda, R., & Hart, P. (1973). *Pattern classification and scene analysis*. John Wiley and Sons.

Dumais, S., Platt, J., Heckerman, D., & Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. *CIKM* (pp. 148–155).

Fellbaum, C. (Ed.). (1998). *Wordnet: An electronic lexical database*. MIT Press.

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

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *ECML'98* (pp. 137–142).

Joachims, T. (1999). Making large-scale SVM learning practical. In B. Schoelkopf, C. Burges and A. Smola (Eds.), *Advances in kernel methods – support vector learning*. The MIT Press.

Lang, K. (1995). Newsweeder: Learning to filter netnews. *ICML'95* (pp. 331–339).

Leopold, E., & Kindermann, J. (2002). Text categorization with support vector machines: How to represent texts in input space. *Machine Learning*, *46*, 423–444.

Lewis, D. D., Yang, Y., Rose, T., & Li, F. (2004). RCV1: A new benchmark collection for text categorization research. *JMLR*, *5*, 361–397.

Mladenic, D. (1998). Feature subset selection in text learning. *ECML'98* (pp. 95–100).

Mladenic, D., & Grobelnik, M. (1998). Word sequences as features in text-learning. *Proc. of 7th Electrotech. and Comp. Sci. Conf.* (pp. 145–148).

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. *EMNLP'02* (pp. 79–86).

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. Morgan Kaufmann.

Reuters (1997). *Reuters-21578 text categorization test collection, Distribution 1.0*. Reuters. `http://www.daviddlewis.com/resources/testcollections/reuters21578`.

Rogati, M., & Yang, Y. (2002). High-performing feature selection for text classification. *CIKM'02* (pp. 659–661).

Scott, S., & Matwin, S. (1999). Feature engineering for text classification. *ICML'99* (pp. 379–388).

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comp. Surveys*, *34*, 1–47.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer-Verlag.

Wermter, S., & Hung, C. (2002). Selforganizing classification on the reuters news corpus. *COLING'02*.

Yang, Y., & Liu, X. (1999). A re-examination of text categorization methods. *SIGIR'99* (pp. 42–49).

Yang, Y., & Pedersen, J. (1997). A comparative study on feature selection in text categorization. *ICML'97* (pp. 412–420).

Yang, Y., Slattery, S., & Ghani, R. (2002). A study of approaches to hypertext categorization. *JIIS*, *18*, 219–241.