Choose Your Words Carefully: An Empirical Study of Feature Selection Metrics for Text Classification

George Forman
Software Technology Laboratory
HP Laboratories Palo Alto

HPL-2002-88 (R.2)
September 20th , 2002*

E-mail: gforman@hpl.hp.com

supervised machine learning, document categorization, support vector machines, binormal separation, residual failure analysis

Good feature selection is essential for text classification to make it tractable for machine learning, and to improve classification performance. This study benchmarks the performance of twelve feature selection metrics across 229 text classification problems drawn from Reuters, OHSUMED, TREC, etc. using Support Vector Machines. The results are analyzed for various objectives. For best accuracy, F-measure or recall, the findings reveal an outstanding new feature selection metric, "Bi-Normal Separation" (BNS). For precision alone, however, Information Gain (IG) was superior. A new evaluation methodology is offered that focuses on the needs of the data mining practitioner who seeks to choose one or two metrics to try that are mostly likely to have the best performance for the single dataset at hand. This analysis determined, for example, that IG and Chi-Squared have correlated failures for precision, and that IG paired with BNS is a better choice.

* Internal Accession Date Only Approved for External Publication

Published in and presented at the 13th European Conference on Machine Learning (ECML '02)/6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), 19-23 August 2002, Helsinki, Finland

Choose Your Words Carefully: An Empirical Study of Feature Selection Metrics for Text Classification

George Forman

Hewlett-Packard Laboratories
1501 Page Mill Rd. MS 1143
Palo Alto, CA, USA 94304
gforman@hpl.hp.com

Abstract. Good feature selection is essential for text classification to make it tractable for machine learning, and to improve classification performance. This study benchmarks the performance of twelve feature selection metrics across 229 text classification problems drawn from Reuters, OHSUMED, TREC, etc. using Support Vector Machines. The results are analyzed for various objectives. For best accuracy, F-measure or recall, the findings reveal an outstanding new feature selection metric, "Bi-Normal Separation" (BNS). For precision alone, however, Information Gain (IG) was superior. A new evaluation methodology is offered that focuses on the needs of the data mining practitioner who seeks to choose one or two metrics to try that are mostly likely to have the best performance for the single dataset at hand. This analysis determined, for example, that IG and Chi-Squared have correlated failures for precision, and that IG paired with BNS is a better choice.

1 Introduction

As online resources continue to grow exponentially, so too will the need to improve the efficiency and accuracy of machine learning methods: to categorize, route, filter and search for relevant text information. Good feature selection can (1) improve classification accuracy—or equivalently, reduce the amount of training data needed to obtain a desired level of performance—and (2) conserve computation, storage and network resources needed for training and all future use of the classifier. Conversely, poor feature selection limits performance—no degree of clever induction can make up for a lack of predictive signal in the input features.

This paper presents the highlights of an empirical study of twelve feature selection metrics on 229 text classification problem instances drawn from 19 datasets that originated from Reuters, OHSUMED, TREC, etc. [3]. (For more details of the study than space permits here, see [1].) We analyze the results from various perspectives, including accuracy, precision, recall and F-measure, since each is appropriate in different situations. Further, we introduce a novel analysis that is focused on a subtly different goal: to give guidance to the data mining practitioner about which feature selection metric or combination is *most likely* to obtain the best performance for the *single given* dataset they are faced with, supposing their text classification problem is drawn from a distribution of problems similar to that studied here.

Our primary focus is on obtaining the best overall classification performance regardless of the number of features needed to obtain that performance. We also analyze which metrics excel for small sets of features, which is important for situations where machine resources are severely limited, low latency classification is needed, or large scalability is demanded.

The results on these benchmark datasets showed that the well-known Information Gain metric was not best for the goals of F-measure, Recall or Accuracy, but instead an outstanding new feature selection metric, "Bi-Normal Separation." For the goal of Precision alone, however, Information Gain was superior.

In large text classification problems, there is typically a substantial skew in the class distribution. For example, in selecting news articles that best match one's personalization profile, the positive class of interest contains many fewer articles than the negative background class. For multi-class problems, the skew increases with the number of classes. The skew of the classification problems used in this study is 1:31 on average, and ~4% exceed 1:100. High class skew presents a particular challenge to induction algorithms, which are hard pressed to beat the high accuracy achieved by simply classifying everything as the negative majority class. For this reason, accuracy scores can under-represent the value of good classification. Precision and recall are often preferable measures for these situations, or their harmonic average, F-measure.

High class skew makes it that much more important to supply the induction algorithm with well chosen features. In this study, we consider each binary class decision as a separate problem instance and select features for it alone. This is the natural setting for 2-class problems, e.g. in identifying spam vs. valuable email. This is also an important subcomponent for good multi-class feature selection [2], i.e. determining a fixed set of features for multiple 2-class problems (aka "n-of-m," *topic or keyword identification*), or for "1-of-m" multi-class problems, e.g. determining where to file a new item for sale in the large Ebay.com classified ad categories.

The choice of the induction algorithm is not the object of study here. Previous studies have shown Support Vector Machines (SVM) to be a consistent top performer [e.g. 6], and a pilot study comparing the use of the popular Naïve Bayes algorithm, logistic regression, and C4.5 decision trees confirmed the superiority of SVM. (When only a small number of features are selected, however, we found Naïve Bayes to be the best second choice, compared to the others.)

Related Work: For context, we mention that a large number of studies on feature selection have focused on non-text domains. These studies typically deal with much lower dimensionality, and often find that wrapper methods perform best. Wrapper methods, such as sequential forward selection or genetic search, perform a search over the space of all possible subsets of features, repeatedly calling the induction algorithm as a subroutine to evaluate various subsets of features. For high-dimensional problems, however, this approach is intractable, and instead feature scoring metrics are used independently on each feature. This paper is only concerned with feature scoring metrics; nevertheless, we note that advances in scoring methods should be welcome to wrapper techniques for use as heuristics to guide their search more effectively.

Previous feature selection studies for *text* domain problems have not considered as many datasets, tested as many metrics, nor considered support vector machines. For example, the valuable study by Yang and Pedersen [7] considered five feature selection metrics on the standard Reuters dataset and OHSUMED. It did not consider SVM, which they later found to be superior to the algorithms they had studied, LLSF and kNN [6]. The question remains then: do their findings generalize to SVM?

Such studies typically consider the problem of selecting one set of features for 1-of-m or n-of-m multi-class problems. This fails to explore the best possible accuracy obtainable for any single class, which is especially important for high class skew. Also, as pointed out in [2], all feature scoring metrics can suffer a blind spot for multi-class problems when there are many good predictive features available for one or a few easy classes that overshadow the useful features for difficult classes.

This study also recommends feature selection strategies for varied situations, e.g. different tradeoffs between precision and recall, and for when resources are tight.

2 Feature Selection Methods

The overall feature selection procedure is to score each potential word/feature according to a particular feature selection metric, and then take the best *k* features. Scoring a feature involves counting its occurrences in training examples for the positive and the negative classes separately, and then computing a function of these.

In addition, there are some other filters that are commonly applied. First, rare words may be eliminated, on the grounds that they are unlikely to be present to aid any given classification. For example, on a dataset with thousands of words, those occurring two or fewer times may be removed. Word frequencies typically follow a Zipf distribution ($\sim 1/\text{rank}^p$). Easily half the total number of unique words may occur only a single time, so eliminating words under a given low rate of occurrence yields great savings. The particular choice of threshold can have an effect on accuracy, and we consider this further in our evaluation. If we eliminate rare words based on a count from the whole dataset *before* we split off a training set, we have leaked some information about the test set to the training phase. Without expending a great deal more resources for cross-validation studies, this research practice is unavoidable, and is considered acceptable in that it does not use the class labels of the test set.

Additionally, overly common words, such as “a” and “of”, may also be removed on the grounds that they occur so frequently as to not be discriminating for any particular class. Common words can be identified either by a threshold on the number of documents the word occurs in, e.g. if it occurs in over half of all documents, or by supplying a *stopword* list. Stopwords are language-specific and often domain-specific. Depending on the classification task, they may run the risk of removing words that are essential predictors, e.g. the word “can” is discriminating between “aluminum” and “glass” recycling.

It is also to be mentioned that the common practice of *stemming* or *lemmatizing*—merging various word forms such as plurals and verb conjugations into one distinct term—also reduces the number of features to be considered. It is properly considered, however, a *feature engineering* option.

An ancillary feature engineering choice is the representation of the feature value. Often a Boolean indicator of whether the word occurred in the document is sufficient. Other possibilities include the count of the number of times the word occurred in the document, the frequency of its occurrence normalized by the length of the document, the count normalized by the inverse document frequency of the word. In situations where the document length varies widely, it may be important to normalize the counts. For the datasets included in this study, most documents are short, and so normalization is not called for. Further, in short documents words are unlikely to repeat, making Boolean word indicators nearly as informative as counts. This yields a great savings in training resources and in the search space of the induction algorithm. It may otherwise try to discretize each feature optimally, searching over the number of bins and each bin's threshold. For this study, we selected Boolean indicators for each feature. This choice also widens the choice of feature selection metrics that may be considered, e.g. Odds Ratio deals with Boolean features, and was reported by Mladenic and Grobelnik to perform well [5].

A final choice in the feature selection policy is whether to rule out all negatively correlated features. Some argue that classifiers built from positive features only may be more transferable to new situations where the background class varies and retraining is not an option, but this benefit has not been validated. Additionally, some classifiers work primarily with positive features, e.g. the Multinomial Naïve Bayes model, which has been shown to be both better than the traditional Naïve Bayes model, and considerably inferior to other induction methods for text classification [e.g. 6]. Negative features are numerous, given the large class skew, and quite valuable in practical experience. For example, when scanning a list of Web search results for the author's home page, a great number of hits on George Foreman the boxer show up and can be ruled out strongly via the words "boxer" and "champion," of which the author is neither. The importance of negative features is empirically confirmed in the evaluation.

2.1 Metrics Considered

Here we enumerate the feature selection metrics we evaluated. In the interest of brevity, we omit the equations and mathematical justifications for the metrics that are widely known (see [1,5,7]). Afterwards, we show a novel graphical analysis that reveals the widely different decision curves they induce. Paired with an actual sample of words, this yields intuition about their empirical behavior.

Notation: P(+) and P(-) represent the probability distribution of the positive and negative classes; *pos* is the number of documents in the positive class. The variables *tp* and *fp* represent the raw word occurrence counts in the positive and negative classes, and *tpr* and *fpr* indicate the sample true-positive-rate, P(word|+), and false-positive-rate, P(word|-). These summary statistics are appropriate for Boolean features. Note that any metric that does not have symmetric values for negatively correlated features is made to value negative features equally well by inverting the value of the feature, i.e. $tpr' = 1 - tpr$ and $fpr' = 1 - fpr$, without reversing the classes.

Commonly Used Metrics:

Chi: Chi-Squared measures the divergence from the expected distribution assuming the feature is actually independent of the class value.

IG: Information Gain measures the decrease in entropy when given the feature. Yang and Pederson reported IG and Chi performed very well [7].

Odds: Odds Ratio reflects the probability ratio of the (positive) class given the feature. In the study by Mladenic and Grobelnik [5] it yielded the best F-measure for Multinomial Naïve Bayes, which works primarily from positive features.

DFreq: Document Frequency simply measures in how many documents the word appears, and can be computed without class labels. It performed much better than Mutual Information in the study by Yang and Pedersen, but was consistently dominated by IG and Chi.

Additional Metrics:

Rand: Random ranks all features randomly and is used as a baseline for comparison. Interestingly, it scored highest for precision in the study [5], although this was not considered valuable because its recall was near zero, yielding the lowest F-measure scores.

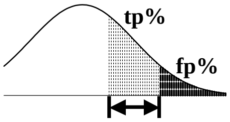
Acc: Accuracy estimates the expected accuracy of a simple classifier built from the single feature, i.e. P(1 for + class and 0 for – class) = P(1|+) P(+) + P(0|-)P(-) = *tpr* P(+) + (1-*fpr*) P(-), which simplifies to the simple decision surface *tp* – *fp*. Note that it takes the class skew into account. Since P(-) is large, *fpr* has a strong influence. When the classes are highly skewed, however, better accuracy can sometimes be achieved simply by always categorizing into the negative class.

Acc2: Accuracy2 is similar, but supposes the two classes were balanced in the equation above, yielding the decision surface *tpr* – *fpr*. This removes the strong preference for low *fpr*.

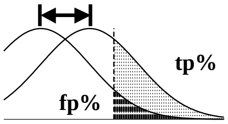
F1: F_1-measure is the harmonic mean of the precision and recall: 2 *recall precision* / (*recall* + *precision*), which simplifies to 2 *tp* / (*pos* + *tp* + *fp*). This metric is motivated because in many studies the F-measure is the ultimate measure of performance of the classifier. Note that it focuses on the positive class, and that negative features, even if inverted, are devalued compared to positive features. This is ultimately its downfall as a feature selection metric.

OddN: Odds Numerator is the numerator of Odds Ratio, i.e. *tpr* * (1-*fpr*).

PR: Probability Ratio is the probability of the word given the positive class divided by the probability of the word given the negative class, i.e. *tpr/fpr*. It induces the same decision surface as log(*tpr/fpr*), which was studied in [5]. Since it is not defined at *fpr*=0, we explicitly establish a preference for features with higher *tp* counts along the axis by substituting *fpr'*=1e-8.



BNS: Bi-Normal Separation is a new feature selection metric we defined as $F^{-1}(tpr) - F^{-1}(fpr)$, where F^{-1} is the standard Normal distribution's inverse cumulative probability function. For intuition, suppose the occurrence of a given feature in each document is modeled by the event of a random Normal variable exceeding a hypothetical threshold. The prevalence rate of the feature corresponds to the area under the curve past the threshold. If the feature is more prevalent in the positive class, then its threshold is further from the tail of the distribution than that of the negative class. The BNS metric measures the separation between these thresholds.



An alternate view is motivated by ROC threshold analysis: The metric measures the horizontal separation between two standard Normal curves where their relative position is uniquely prescribed by *tpr* and *fpr*, the area under the tail of each curve (cf. a traditional hypothesis test where *tpr* and *fpr* estimate the center of each curve). The BNS distance metric is therefore proportional to the area under the ROC curve generated by the two overlapping Normal curves, which is a robust method that has been used in the medical testing field for fitting ROC curves to data in order to determine the efficacy of a treatment. Its justifications in the medical literature are many and diverse, both theoretical and empirical [4].

Pow: Pow is $(1\text{-}fpr)^k - (1\text{-}tpr)^k$, where k is a parameter. Although theoretically unmotivated, it is considered because it prefers frequent terms [7], aggressively avoids common *fp* words, and can generate a variety of decision surfaces given parameter k, with higher values corresponding with a stronger preference for positive words. This leaves the problem of optimizing k. We chose k=5 after a pilot study.

2.2 Graphical Analysis

In order to gain a more intuitive grasp for the selection biases of these metrics, we present in Figure 1 the actual decision curves they induce in ROC space—true positives vs. false positives—when selecting exactly 100 words for distinguishing abstracts of general computer science papers vs. those on probabilistic machine learning techniques. The horizontal axis represents far more negative documents (1750) than the vertical axis (50), for a skew of 1:35. The triangle below the diagonal represents negatively correlated words, and the symmetrically inverted decision curves are shown for each metric. We see that Odds Ratio and BNS treat the origin and upper right corner equivalently, while IG and Chi progressively cut off the top right—and symmetrically the bottom left, eliminating many negative features.

The dots represent the specific word features available in this problem instance—note that there are many words sharing the same *tp* and *fp* counts near the origin, but the black and white visualization does not indicate the many collisions. Very few words have high frequency and they also tend to be non-predictive, i.e. they stay close to the diagonal as they approach the upper right corner. This partly supports the practice of eliminating the most frequent words (the bold dotted line depicts a cut-off threshold that eliminates words present in >¼ of all documents), but note that it saves only 28 words out of 12,500.

Since word frequencies tend toward a Zipf distribution, most of the potential word features appear near the origin. This implies that feature selection is most sensitive to the shape of the decision curve in these dense regions. Figure 2 shows a zoomed-in view where most of the words occur. The bold diagonal line near the origin shows a rare word cutoff of <3 occurrences, which eliminates 7333 words for this dataset. This represents substantial resource savings (~60%) and the elimination of fairly uncertain words that are unlikely to re-occur at a rate that would be useful for classification.

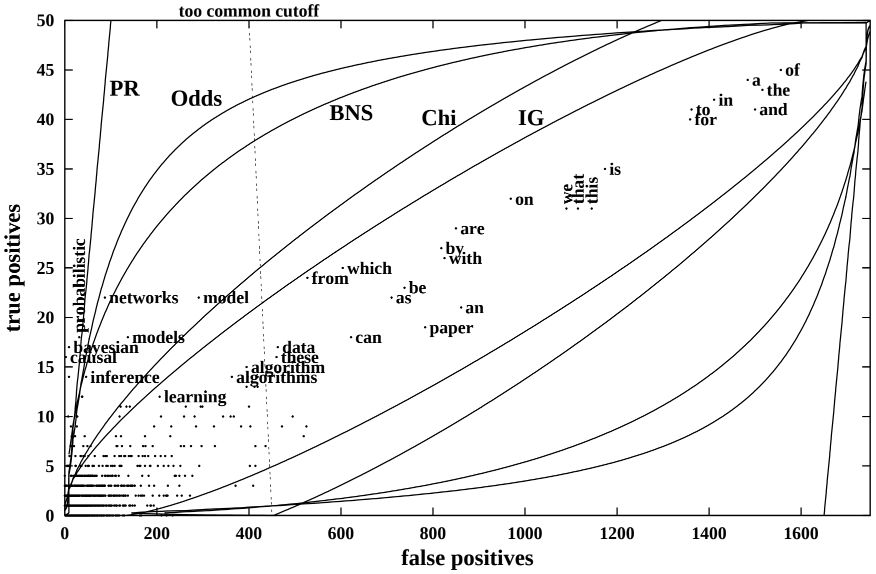


Fig. 1. Decision boundary curves for the feature selection metrics Probability Ratio, Odds Ratio, Bi-Normal Separation, Chi-Squared, and Information Gain. Each curve selects the "best" 100 words, each according to its view, for discriminating abstracts of data mining papers from others. Dots represent actual words, and many of the 12K words overlap near the origin.

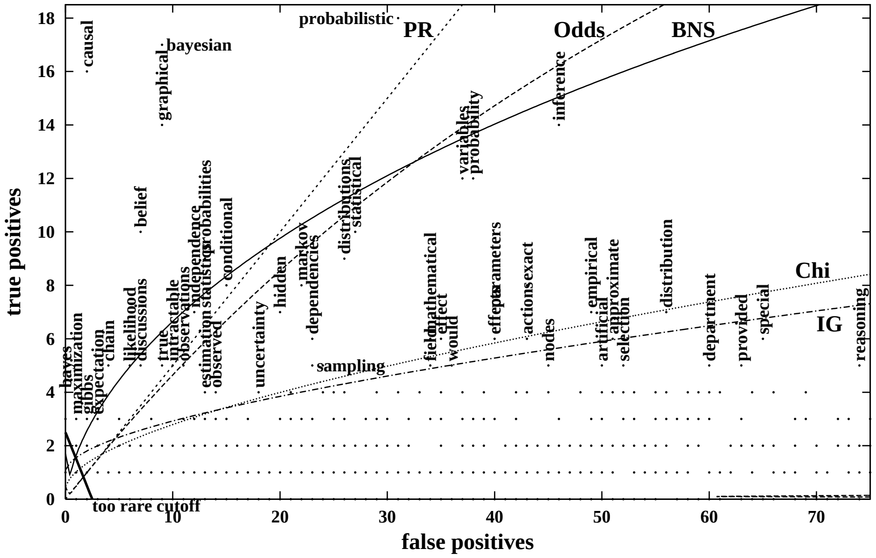


Fig. 2. Zoomed-in version of Figure 1, detailing where most words occur.

3 Experimental Method

Performance measures: While several studies have sought solely to maximize the F-measure, there are common situations where precision is to be strongly preferred over recall, e.g. when the cost of false positives is high, such as mis-filtering a legitimate email as spam. Precision should also be the focus when delivering Web search results, where the user is likely to look at only the first page or two of results. Finally, there are situations where accuracy is the most appropriate measure, even when there is high class skew, e.g. equal misclassification costs. For these reasons, we analyze the performance for each of the four performance goals.

There are two methods for averaging the F-measure over a collection of 2-class classification problems. One is the *macro-averaged F-measure*, which is the traditional arithmetic mean of the F-measure computed for each problem. Another is the *micro-averaged F-measure*, which is an average weighted by the class distribution. The former gives equal weight to each problem, and the latter gives equal weight to each document classification (which is equivalent to overall accuracy for a 1-of-m problem). Since highly skewed, small classes tend to be more difficult, the macro-averaged F-measure tends to be lower. We focus on macro-averaging because we are interested in average performance across different problems, without regard to the problem size of each. (To measure performance for a given problem instance, we use 4-fold stratified cross-validation, and take the average of 5 runs.)

A data mining practitioner has a different goal in mind—to choose a feature selection technique that maximizes their *chances* of having the best metric *for their single dataset of interest.* Supposing the classification problems in this study are representative of problems encountered in practice, we compute for each metric, the percentage of problem instances for which it was optimal, or within a given error tolerance of the best method observed for that instance.

Induction Algorithm: We performed a brief pilot study using a variety of classifiers, including Naïve Bayes, C4.5, logistic regression and SVM with a linear kernel (each using the WEKA open-source implementation with default parameters). The results confirmed previous findings that SVM is an outstanding method [6], and so the remainder of our presentation uses it alone. It is an interesting target for feature selection because no comparative text feature selection studies have yet considered it, and its use of features is entirely along the decision boundary between the positive and negative classes, unlike many traditional induction methods that model the density. We note that the traditional Naïve Bayes model fared better than C4.5 for these text problems, and that it was fairly sensitive to feature selection, having its performance peak at a much lower number of features selected.

Datasets: We were fortunate to obtain a large number of text classification problems in preprocessed form made available by Han and Karypis, the details of which are laid out in [3] and in the full version of this study [1]. These text classification problems are drawn from the well known Reuters, OHSUMED and TREC datasets. In addition, we included a dataset of abstracts of computer science papers gathered from Cora.whizbang.com that were categorized into 36 classes, each containing 50 training examples. Taken altogether, these represent 229 two-class text classification problem instances, with a positive class size of 149 on average, and class skews averaging 1:31 (median 1:17, 5th percentile 1:3, 95th 1:97, max 1:462).

Feature Engineering and Selection: Each feature represents the Boolean occurrence of a forced-lowercase word. Han [3] reports having applied a stopword list and Porter's suffix-stripping algorithm. From an inspection of word counts in the data, it appears they also removed rare words that occurred <3 times in most datasets. Stemming and stopwords were not applied to the Cora dataset, and we used the same rare word threshold. We explicitly give equal importance for negatively correlated word features by inverting *tpr* and *fpr* before computing the feature selection metric. We varied the number of selected features in our experiments from 10 to 2000. Yang and Pedersen evaluated up to 16,000 words, but the F-measure had already peaked below 2000 for Chi-Squared and IG [7]. If the features are selected well, most of the information should be contained in the initial features selected.

4 Empirical Results

Figure 3 shows the macro-averaged F-measure for each of the feature selection metrics as we vary the number of features to select. The absolute values are not of interest here, but rather the overall trends and the separation of the top performing curves. We see that to maximize the F-measure on average, BNS performed best by a wide margin, using 500 to 1000 features. This is a significant result in that BNS has not been used for feature selection before, and the significance level, even in the barely visible gap between BNS and IG at 100 features, is greater than 99.9% confidence in a paired t-test of the 229*5 runs. Like the results of Yang and Pedersen [7], performance begins to decline around 2000 features.

If for scalability reasons one is limited to 20-50 features, a better metric to use is IG (or Acc2, which is simpler to program. Surprisingly, Acc2, which ignores class skew, performs much better than Acc, which accounts for skew.). IG dominates the performance of Chi at every size of feature set.

Accuracy: The results for accuracy are much the same, and their graphs must be omitted for space (but see [1]). BNS again performed the best by a smaller, but still >99.9% confident, margin. At 100 features and below, however, IG performed best, with Acc2 being statistically indistinguishable at 20 features.

Precision-Recall Tradeoffs: As discussed, one's goal in some situations may be solely precision or recall, rather than F-measure. Figure 4 shows this tradeoff for each metric, macro-averaged across all sample problems and evaluated at 1000 features selected. We see that the success of BNS with regard to its high F-measure is because it obtains on average much higher recall than any other method. If, on the other hand, precision is the sole goal, IG is the best at any number of features (and Chi is statistically indistinguishable over 1000 features).

4.1 Best Chances of Obtaining Maximum Performance

The problem of choosing a feature selection metric is somewhat different when viewed from the perspective of a data mining practitioner whose task is to get the best performance on a *given* set of data, rather than averaging over a large number of datasets. Practitioners would like guidance as to which metric is most *likely* to yield

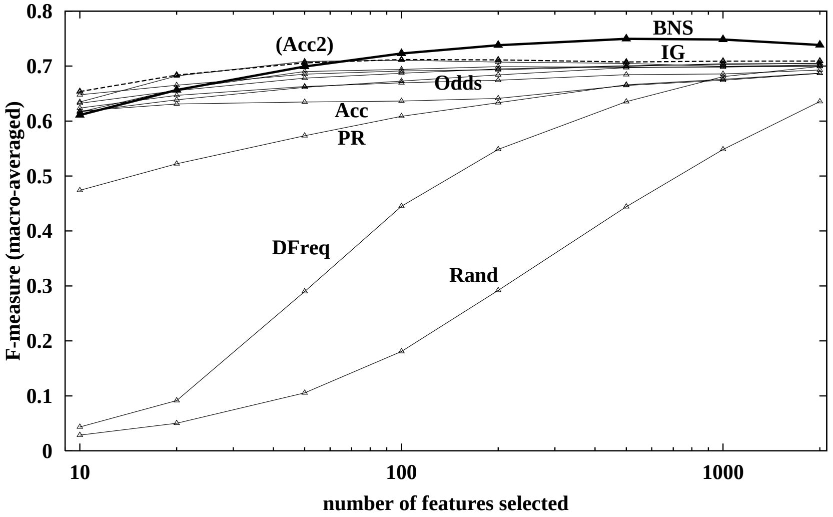


Fig. 3. F-measure averaged over 229 problems for each metric & number of features.

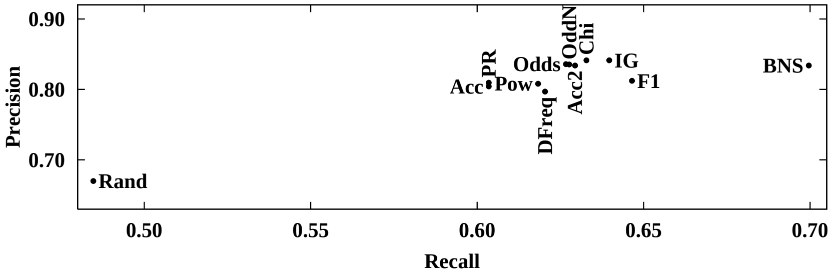


Fig. 4. Precision-Recall tradeoffs at 1000 features from Fig. 3.

the best performance for their single dataset at hand. Supposing the problem instance is drawn from a distribution similar to that in this study, we offer the following analysis: For each feature selection metric, we determine the percentage of the 229 problem instances for which it matched the best performance found within a small tolerance (taking the maximum over any number of features). We repeat this separately for F-measure, precision, recall and accuracy.

Figure 5a shows these results for the goal of maximum F-measure as we vary the acceptable tolerance from 1% to 10%. As it increases, each metric stands a greater chance of obtaining close to the maximum, thus the trend. We see that BNS attained within 1% of best performance for 65% of the 229 problems, beating IG at just 40%. Figure 5b shows similar results for Accuracy, F-measure, Precision and Recall (but using 0.1% tolerance for accuracy, since large class skew compresses the range). Note

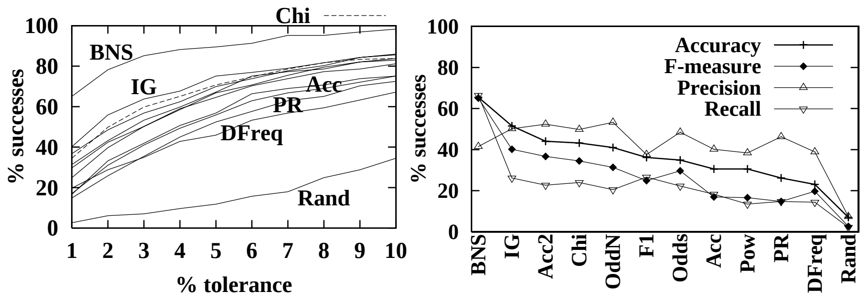


Fig. 5. (a) Percentage of problems on which each metric scored within *x*% tolerance of the best F-measure of any metric. (b) Same, for F-measure, recall, and precision @1%, accuracy @0.1%.

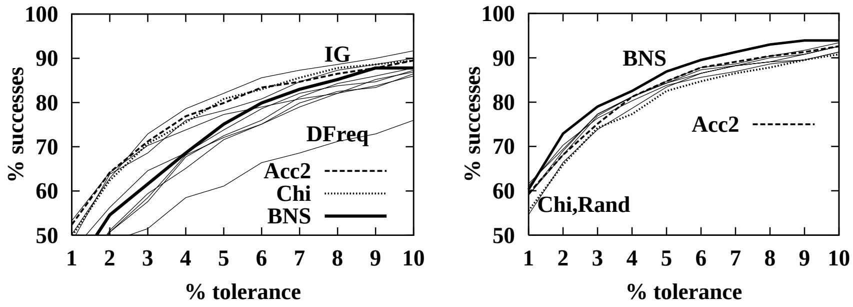


Fig. 6. (a) As Figure 5a, but for precision. (b) Same, but each metric is combined with IG.

that for precision, several metrics beat BNS, notably IG. This is seen more clearly in Figure 6a, which shows these results for varying tolerances. IG consistently dominates at higher tolerances, though the margin is less striking than Figure 5a.

Residual Analysis: If one were willing to invest the extra effort to try two different metrics for one's dataset and select the one with better precision via cross-validation, the two leading metrics, IG and Chi, would seem a logical choice. However, it may be that wherever IG fails to attain the maximum, Chi also fails. To evaluate this, we repeated the procedure considering the maximum performance of pairs of feature selection metrics. Figure 6b shows these results for each metric paired with IG. Observe that, surprisingly, IG+Chi performed the worst of the pairs, validating the hypothesis that it has correlated failures. BNS, on the other hand, has uncorrelated failures and so, paired with IG, gives the best precision, and by a significant margin.

This paired analysis was repeated for F-measure, Recall and Accuracy, and consistently revealed that BNS paired with IG sustained the best performance.

Due to space limitations, refer to [1] for the complete results, as well as related experiments we performed. Below we briefly mention two of the ancillary findings:

Lesion Study on Negative Features: In the experiments above, we inverted negative features so that they would be treated identically to positive features, creating the symmetrical decision surfaces seen in Figure 1. In a related suite of experiments, we suppressed negative features altogether to determine their importance. When deprived of negative features, no feature selection metric was competitive with the previous results for BNS, IG or Acc2. We conclude that negative features are essential to high quality classification.

Sensitivity to the Rare Word Cutoff: In the existing datasets, words were removed that occurred fewer than 3 times in the (training & testing) corpus. Some text preparation practices use a much higher threshold to reduce the size of the data. We performed a suite of experiments on the Cora dataset, varying this threshold up to 25, which eliminates the majority of the potential features from which to select. The macro-averaged F-measure, precision and accuracy (for BNS at 1000 features selected) each decline steadily as the threshold increases; recall rises slightly up to a theshold of 10, and then falls off. We conclude that to reduce space, one should set the rare word cutoff low, and then perform aggressive feature selection using a metric.

5 Conclusion

This paper presented an extensive comparative study of feature selection metrics for the text domain, focusing on support vector machines and 2-class problems, typically with high class skew. It revealed an outstanding new feature selection metric, Bi-Normal Separation, which is mathematically equivalent to the bi-normal assumption that has been used in the medical field for fitting ROC curves [4]. Another contribution of this paper is a novel evaluation methodology that considers the common problem of trying to select one or two metrics that have the best chances of obtaining the best performance for a *given* dataset. Somewhat surprisingly, selecting the two best performing metrics is sub-optimal, because when the best metric fails, the other may have correlated failures, as occurs for IG and Chi. Pairing IG with BNS is consistently a better choice. The reader is referred to [1] for additional results.

Acknowledgements

We wish to thank the WEKA project for their software, the ID Laboratory/INRIA for use of their hardware, and Han & Karypis and Tom Fawcett for the prepared datasets.

References

1. Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Tech Report HPL-2002-147, Hewlett-Packard Laboratories. Submitted to Special Issue on Variable and Feature Selection, J. of Machine Learning Research. (2002)
2. Forman, G.: Avoiding the Siren Song: Undistracted Feature Selection for Multi-Class Text Classification. TR HPL-2002-75, Hewlett-Packard Laboratories. Submitted as above. (2002)
3. Han, E.S., Karypis, G.: Centroid-Based Document Classification: Analysis & Experimental Results. In: Principles of Data Mining and Knowledge Discovery (PKDD). (2000) 424-431
4. Hanley, J.A.: The Robustness of the “Binormal” Assumptions Used in Fitting ROC Curves. Medical Decision Making, 8(3). (1988) 197-203
5. Mladenic, D., Grobelnik, M.: Feature Selection for Unbalanced Class Distribution and Naïve Bayes. In: 16th International Conference on Machine Learning (ICML). (1999)
6. Yang, Y., Liu, X.: A Re-examination of Text Categorization Methods. In: ACM SIGIR Conference on Research and Development in Information Retrieval. (1999) 42-49
7. Yang, Y., Pedersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In: International Conference on Machine Learning (ICML). (1997) 412-420