Weighted Average Pointwise Mutual Information for Feature Selection in Text Categorization

Karl-Michael Schneider

Department of General Linguistics, University of Passau,
94030 Passau, Germany
schneide@phil.uni-passau.de

Abstract. Mutual information is a common feature score in feature selection for text categorization. Mutual information suffers from two theoretical problems: It assumes independent word variables, and longer documents are given higher weights in the estimation of the feature scores, which is in contrast to common evaluation measures that do not distinguish between long and short documents. We propose a variant of mutual information, called *Weighted Average Pointwise Mutual Information* (WAPMI) that avoids both problems. We provide theoretical as well as extensive empirical evidence in favor of WAPMI. Furthermore, we show that WAPMI has a nice property that other feature metrics lack, namely it allows to select the best feature set size automatically by maximizing an objective function, which can be done using a simple heuristic without resorting to costly methods like EM and model selection.

1 Introduction

Automatic text categorization, i.e. the assignment of text documents to predefined categories, is an important task in many NLP applications. The common *bag of words* approach results in a document space with very high dimensionality. In order to speed up parameter estimation and classification and to improve the classifier performance, it is common to use feature selection to reduce the dimensionality of the document space. This is typically done using a filtering approach [1] in which each feature is assigned a score based on an independent evaluation, and the features are then ranked according to their scores, and the N highest ranked features are selected, where N is the desired vocabulary size. Wrapper methods, which use the classifier directly to evaluate different feature subsets [1], are not commonly used for text classification because of the high dimensionality of the feature space that makes searching for the best feature subset intractable.

Mutual Information (MI) is an information-theoretic measure that is often used to evaluate features. It measures the amount of information that the value of a feature in a document (e.g. the presence or absence of a word) gives about the class of the document. Feature selection studies have obtained good results with MI [2]. However, there are two problems associated with the use of MI for feature ranking: First, MI treats each feature as an independent random variable. This is a problem because words in a text are not independent. Second, classifiers based on generative models, such as Naive

A. Jorge et al. (Eds.): PKDD 2005, LNAI 3721, pp. 252–263, 2005.

Bayes [3], estimate class-conditional probability distributions over words from training data. In the multinomial Naive Bayes model [3,4] this is done by concatenating the training documents in each class to one long document and estimating the distribution of words in this long document. This gives larger weights to longer documents. However, in classifier evaluation, all (test) documents have equal weight irrespective of their length—that is, there is a mismatch between classifier training and evaluation.

This paper proposes a variant of MI, *Weighted Average Pointwise Mutual Information* (WAPMI) that avoids both aforementioned problems. We present theoretical (using an information-theoretic argument that links WAPMI to multinomial Naive Bayes) and empirical evidence (through extensive experimentation) in favor of WAPMI. WAPMI improves the performance of multinomial Naive Bayes over MI on a variety of standard benchmark corpora. It also outperforms several other standard metrics for feature ranking.

In addition, WAPMI has a very nice property compared to other metrics, including MI: It allows to determine the (theoretically) best feature set size by maximizing an objective function. This can be done using a simple heuristic by applying a general, data-independent threshold to the feature scores, without the need to resort to computationally intensive methods like EM and model selection. Other feature metrics only evaluate the relative usefulness, and it is not entirely clear how they could be used to define an objective function for feature selection.

We demonstrate the effectiveness of this general thresholding method in our experiments. On some datasets (notably those that are commonly regarded "easy" classification tasks) we obtain smaller feature sets and better performance, while on "difficult" datasets (i.e. large datasets with great variability in the vocabulary) WAPMI selects larger feature sets than other metrics while outperforming them.

The paper is structured as follows. In Sect. 2 we review the probabilistic framework of multinomial Naive Bayes. In Sect. 3 we define weighted average pointwise mutual information and motivate its use for feature ranking. We also discuss its relation to distributional clustering. The experimental setup is described in Sect. 4, and Sect. 4 presents our experiments and the results. Section 5 finishes with some conclusions.

2 Naive Bayes

Naive Bayes is a simple probabilistic classifier that is widely used for text classification [3,4]. Despite this independence assumption, Naive Bayes performs surprisingly well on text classification problems [5].

Let $C = \{c_1, \ldots, c_{|C|}\}$ denote the set of possible classes of documents, and let $V = \{w_1, \ldots, w_{|V|}\}$ be a vocabulary. The multinomial Naive Bayes classifier assumes that a document d is drawn from a multinomial distribution by $|d|$ independent trials on a random variable $W \in V$ with class-conditional distribution $p(w_t|c_j)$ (where $|d|$ denotes document length):

$$p(d|c_j) = p(|d|)|d|! \prod_{t=1}^{|V|} \frac{p(w_t|c_j)^{x_t}}{x_t!}$$

x_t is the number of times W yields w_t, i.e. the number of times the word w_t occurs in d. The parameters $p(w_t|c_j)$ are usually estimated from training documents using maximum likelihood with Laplace smoothing to avoid zero probabilities:

$$\hat{p}(w_t|c_j) = \frac{1+n(c_j,w_t)}{|V|+n(c_j)}$$

where $n(c_j,w_t)$ is the number of occurrences of w_t in the training documents in c_j and $n(c_j)$ is the total number of word occurrences in c_j.

The posterior probability of the class given the document is given by Bayes' rule:

$$p(c_j|d) = \frac{p(c_j)p(d|c_j)}{p(d)}$$

where $p(d)$ is the total probability of d:

$$p(d) = \sum_{j=1}^{|C|} p(c_j)p(d|c_j)$$

The class priors $p(c_j)$ are estimated from training documents as the fraction of documents in class c_j. Given a document, the Naive Bayes classifier selects the class with the highest posterior probability (we can omit those parts that do not depend on the class in the maximization):

$$c^*(d) = \arg\max_{c_j} p(c_j)p(d|c_j) \tag{1}$$

3 Weighted Average Pointwise Mutual Information

3.1 Defining Weighted Average Pointwise Mutual Information

Mutual Information is a measure of the information that one random variable gives about the value of another random variable [6]. Let W be a random variable that ranges over the vocabulary V, and let C be random variable that ranges over classes. The mutual information between W and C is defined as:

$$I(W;C) = \sum_{t=1}^{|V|}\sum_{j=1}^{|C|} p(w_t,c_j)\log\frac{p(w_t|c_j)}{p(w_t)} \tag{2}$$

The term $\log\frac{p(w_t|c_j)}{p(w_t)}$ is called *pointwise mutual information* [7].[1] Note that mutual information can be written as a weighted sum of Kullback-Leibler (KL) divergences. The KL-divergence between two probability distributions p and q is defined as $D(p\|q) = \sum_x p(x)\log\frac{p(x)}{q(x)}$ [6]. Thus (2) can be written as the weighted average KL-divergence

[1] In [2] this is called *information gain*, and the term *mutual information* is used as a synonym for pointwise mutual information.

between the class-conditional distribution of words and the global (unconditioned) distribution in the entire corpus:

$$I(W;C) = \sum_{j=1}^{|C|} p(c_j) D(p(W|c_j) \| p(W))$$

To rank features we would like a measure for each feature. A common method is to define new binary random variables, W_t, for each word that indicate whether the next word in a document is w_t (or some other word) [3,8]: $p(W_t = 1) = p(W = w_t)$. Then the MI-score for w_t is given by:

$$MI(w_t) := I(W_t;C) = \sum_{j=1}^{|C|} \sum_{x=0,1} p(W_t = x, c_j) \log \frac{p(W_t = x|c_j)}{p(W_t = x)} \tag{3}$$

The problem with (3) is that it treats W_t as an independent random variable, but in fact $\sum_{t=1}^{|V|} p(W_t = 1) = 1$! To avoid this independence assumption, we consider (2) as a sum over word scores, where the score for w_t is the pointwise mutual information with the class, averaged over all classes:

$$PMI(w_t) := \sum_{j=1}^{|C|} p(w_t, c_j) \log \frac{p(w_t|c_j)}{p(w_t)} \tag{4}$$

The problem with (4) is that it treats all training documents in one class as one big document (because of the way the class-conditional probabilities are estimated). Thus, if there is variation in the document lengths, (4) is dominated by the longer documents. To avoid this problem, we replace the weight $p(w_t, c_j)$ with a term that is a weighted average of the document-conditional probabilities $p(w_t|d_i) = n(w_t, d_i)/|d_i|$ where $n(w_t, d_i)$ is the number of times w_t occurs in d_i and $|d_i|$ is the length of d_i.[2] Thus weighted average pointwise mutual information is defined as:

$$WAPMI(w_t) := \sum_{j=1}^{|C|} \sum_{d_i \in c_j} \alpha_i p(w_t|d_i) \log \frac{p(w_t|c_j)}{p(w_t)} \tag{5}$$

We consider several alternatives for the weights α_i, which can be associated with different measures for classifier evaluation:

- $\alpha_i = p(c_j) \cdot |d_i| / \sum_{d_i \in c_j} |d_i|$. This gives each document a weight proportional to its lengths and yields (4).
- $\alpha_i = 1/\sum_{j=1}^{|C|} |c_j|$. This gives equal weight to all documents. This corresponds to an evaluation measure that counts each misclassified document as the same error, i.e. classification accuracy.
- $\alpha_i = 1/(|c_j| \cdot |C|)$ where $d_i \in c_j$. This gives equal weight to the classes by normalizing for class size, i.e. documents from smaller categories receive higher weights. This compensates for the dominance of larger categories in classifier evaluation.

[2] Note that any word that does not occur in d_i has zero probability.

By summing (5) over all words we obtain the total weighted average pointwise mutual information between the word variable W and the class variable C:

$$WAPMI(W;C) := \sum_{t=1}^{|V|} \sum_{j=1}^{|C|} \sum_{d_i \in c_j} \alpha_i p(w_t|d_i) \log \frac{p(w_t|c_j)}{p(w_t)} \tag{6}$$

In the following subsections we provide theoretical evidence that total WAPMI could be used as an objective function, and the goal of feature selection is to maximize that objective function.

3.2 Relation to Distributional Clustering

Note that (6) can be written as a weighted sum of the difference between (i) the KL-divergence of the document-conditional distribution from the corpus distribution and (ii) the KL-divergence of the document-conditional distribution from the class-conditional distribution:

$$\sum_{j=1}^{|C|} \sum_{d_i \in c_j} \alpha_i \Big[D(p(W|d_i) \| p(W)) - D(p(W|d_i) \| p(W|c_j)) \Big] \tag{7}$$

This can be interpreted as an estimate of how similar the documents in one class are and how dissimilar documents of different classes are. From a clustering perspective we can say that (7) is large if the documents that belong to the same class form tight clusters, with wide separation between the clusters. Interpreting text categorization as an information retrieval task (i.e. regarding classes as queries) this is a desirable property that has been argued to improve document retrieval performance in the vector space model [9].

In distributional clustering the goal is to cluster similar objects (e.g. documents) together so as to maximize the value of an objective function that measures the quality of the clustering [10]. Below we argue that maximizing (7) is expected to improve the accuracy of the multinomial Naive Bayes classifier. Thus we can regard total weighted average pointwise mutual information as an objective function (since it is a function of the entire training corpus). However, in contrast to clustering, we do not change the clusters (which correspond to the classes in the training corpus and which we consider to be fixed). Instead our goal is to improve the clustering by changing the document representation (i.e. by using a subset of the features).

3.3 Relation to Multinomial Naive Bayes

We can use (7) to get an estimate of the expected performance of Naive Bayes on the training set (and by generalization also on a test set, if the test documents are draw from the same distribution). We manipulate the Naive Bayes classifier (1) in an information theoretic framework using the fact that a document defines a probability distribution over words. We define the distance of a document, d_i, from a class, c_j, as the KL-divergence between the document-conditional word distribution and the class-conditional distribution. Naive Bayes can then be written in the following form by taking logarithms, dividing by the length of d_i and adding the entropy of d_i, $H(p(W|d_i)) = -\sum_t p(w_t|d_i) \log p(w_t|d_i)$ [10]:

$$c^*(d_i) = \arg\min_{c_j} \left[D(p(W|d_i) \| p(W|c_j)) - \frac{1}{|d_i|} \log p(c_j) \right] \quad (8)$$

Note that the modifications in (8) do not change the classification of documents. Assuming equal class priors, Naive Bayes can thus be interpreted as selecting the class which has the least distance from the document. Taking into account the arguments from the previous subsection, maximizing the total weighted average pointwise mutual information (6) would thus increase the probability that each document is nearer to its true class than to any other class, and would therefore be classified correctly by multinomial Naive Bayes.

3.4 Using WAPMI as an Objective Function for Feature Selection

Taking into account the arguments in the previous subsections, the best feature set would be one that maximizes the total WAPMI (6). Note that the WAPMI score (5) can be negative, which suggests the following simple heuristic for maximizing total WAPMI: Simply select all words with a positive WAPMI score and removing all other words. This is equivalent to applying a threshold of $\theta = 0$ to the WAPMI score. We examine this empirically in Sect. 4. In contrast, mutual information is always non-negative (and almost always positive), and it is not entirely clear how mutual information could be used as an objective function in feature selection.

Note that the above heuristic is only an approximation. In fact, feature selection isn't entirely well-defined in multinomial Naive Bayes, since we are not only pruning the model but the data too! Pruning the vocabulary changes the distribution of the remaining words. An alternative would be to not greedily discard words but perform several iterations and recompute the objective function after each iteration until convergence. We tried this, but there was almost no difference. In most cases, convergence occurred after only two or three iterations, with only a few additional words removed after the first round.

4 Experiments

4.1 Datasets and Procedures

We perform experiments on five text categorization datasets, described in Table 1. The 20 Newsgroups dataset[3] consists of Usenet articles distributed evenly in 20 different newsgroups that make up the classes [11]. We remove newsgroup headers and binary attachments and use only words consisting of alphabetic characters as tokens, after converting to lower case and mapping numbers, URLs and email addresses to special tokens.

The WebKB dataset and the 7 Sectors dataset are both available from the WebKB project [12].[4] WebKB contains web pages gathered from computer science departments and categorized in six classes plus one *other* class. We use only the four most populous classes *course*, *faculty*, *project* and *student*. The 7 Sectors data consists of web pages

[3] http://people.csail.mit.edu/people/jrennie/20Newsgroups/

[4] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb/

Table 1. Corpus statistics. The last two columns show the number of documents in the smallest and biggest categories, respectively.

Dataset	Classes	Vocabulary	Documents	Smallest	Largest
20 Newsgroups	20	94,897	19,997	997	1,000
WebKB	4	41,015	4,199	504	1,641
7 Sectors	48	42,110	4,582	39	105
Reuters-10 (train)	10	22,430	6,490	181	2,877
Reuters-10 (test)	10	13,849	2,545	56	1,087
Reuters-90 (train)	90	24,719	7,770	1	2,877
Reuters-90 (test)	90	15,660	3,019	1	1,087

from different companies divided into a hierarchy of classes. We use the flattened version of the data. We strip all HTML tags and use only words and numbers as tokens, after converting to lower case and mapping numbers and other expressions to special tokens.

The Reuters-21578 dataset[5] consists of Reuters news articles belonging to zero or more topic classes. We use the ModApte split [13] and produce two versions of the corpus. Reuters-10 uses only the 10 largest topics. On average, each document belongs to 1.105 topic classes. Reuters-90 uses all 90 topics that have at least one document in the training and test set, with an average of 1.235 topics per document.

Except on Reuters, all experiments are performed using cross-validation. We follow the methodology in [3]. For 20 Newsgroups and 7 Sectors, we split the data into five parts of equal size and with equal class distribution. For WebKB we produce ten train/test splits using stratified random sampling with 70% training and 30% test data. We report average classification accuracy across trials.

For the Reuters experiments we build a binary classifier for each topic, using the documents belonging to each topic as positive examples and all other documents as negative examples. Following the standard methodology with multi-label datasets, we ignore the classification decision of the classifier and use the classification scores to rank the documents. We then report precision/recall breakeven points averaged over all topics (called "macroaverage"). Instead of the Naive Bayes posterior probabilities, which tend to produce extreme values with growing document length due to the Naive Bayes independence assumption and are not comparable across documents, we use the normalized KL-divergence based classification scores described in [12].

4.2 Quality of Selected Features

We compare our WAPMI scoring function against three other scoring functions: Mutual Information [3], Chi-squared [2] and Bi-normal separation [14]. We evaluate the quality of the selected features by varying the number of selected features. We use WAPMI with equal weighting for all documents (we also experimented with equal class weights but found no statistically significant difference). Table 2 shows the top 20 words in the entire 20 Newsgroups corpus according to Mutual Information and WAPMI.

Figure 1 shows classification accuracy on the three datasets. As can be seen, the WAPMI scoring function yields higher classification accuracy, although on WebKB

[5] http://www.daviddlewis.com/resources/testcollections/reuters21578/

Table 2. 20 words with highest MI (left) and WAPMI score (right) in the 20 Newsgroups corpus

MI	Word	MI	Word	WAPMI	Word	WAPMI	Word
0.02833	ax	0.00174	g	0.00221	rainbowthreedigit	0.00073	rainbowdigits
0.01555	rainbowonedigit	0.00168	w	0.00179	sale	0.00070	mac
0.00387	rainbowdigits	0.00161	m	0.00150	rainbowtwodigit	0.00068	clipper
0.00374	rainbowtwodigit	0.00155	u	0.00140	windows	0.00067	taggedemail
0.00336	x	0.00144	v	0.00129	x	0.00067	card
0.00222	q	0.00143	of	0.00091	car	0.00066	thanks
0.00188	rainbowthreedigit	0.00124	god	0.00089	god	0.00065	team
0.00182	f	0.00119	r	0.00087	game	0.00064	he
0.00181	max	0.00109	p	0.00083	drive	0.00064	i
0.00175	the	0.00104	that	0.00074	bike	0.00064	space

the difference is statistically significant only for up to 2,000 words. In general, the improvement seems to be higher on smaller vocabulary sizes.

The class distribution is highly skewed in the Reuters datasets. The largest category (earn) has 2,877 documents in the training set, while the smallest category in Reuters-10 (corn) has 181 documents in the training set. In Reuters-90 there are 29 categories with less than 10 documents in the training set.

For the Reuters experiments we use two versions of WAPMI: with equal weights for all documents (WAPMI1), and with equal class weights (WAPMI2) (cf. Sect. 3.1), which deemphasizes the impact of the larger classes. Figure 2 shows the results on the Reuters datasets with 10 and 90 categories. We report macroaveraged precision/recall breakeven, which gives equal weight to the performance on each category. WAPMI with equal weights on documents does not perform better than the other metrics, except for very small vocabularies on Reuters-90. However, when the weights are set such that documents from smaller categories receive higher weights (WAPMI2), WAPMI clearly outperforms the other feature scoring methods.

4.3 Global Thresholding

In addition to the experiments with varying numbers of features we also examined the possibility of using a global thresholding strategy, with a fixed threshold that is applied to all datasets. We are interested in the sensitivity of the various feature scoring functions to the difficulty of the classification task. In general, the Naive Bayes classifier performs better with large vocabularies, but the optimal vocabulary size depends on the dataset. For instance, the 20 Newsgroups dataset requires a larger vocabulary for optimal classification accuracy than the other datasets [3].

For Mutual Information, Chi-squared and Bi-normal separation we select a threshold that yields relatively good performance on all datasets. For WAPMI we use the theoretically best threshold 0. For all datasets except 20 Newsgroups we use both variants with equal weights on documents (WAPMI1) and on classes (WAPMI2). For 20 Newsgroups WAPMI1 and WAPMI2 are the same because all classes have the same number of documents.

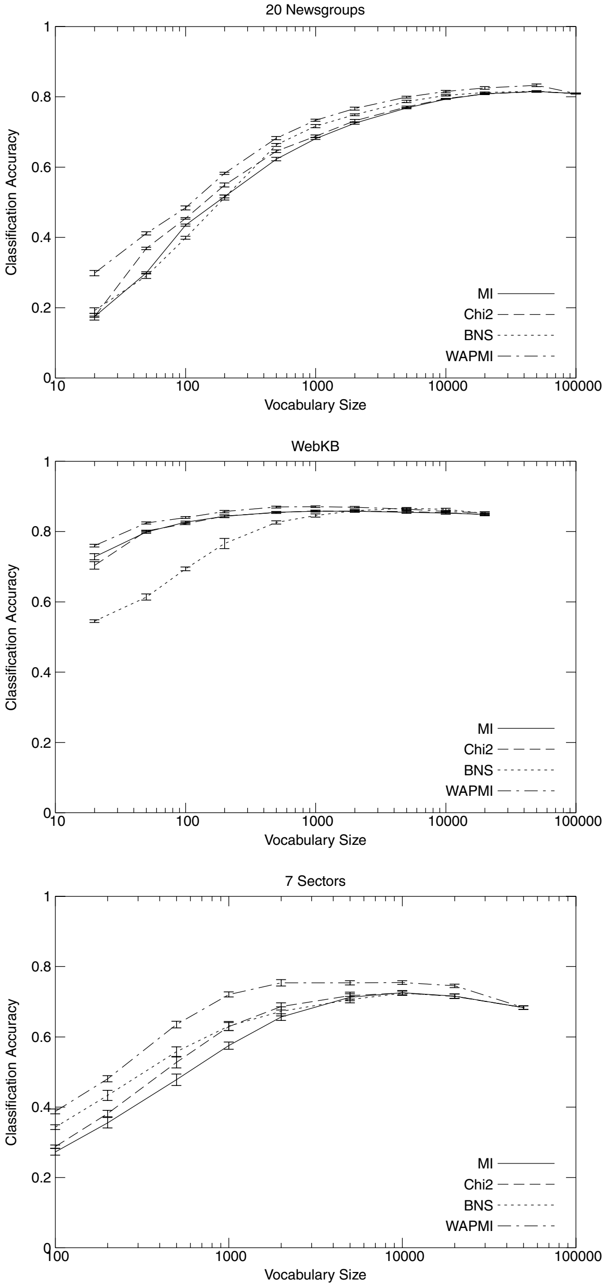

Fig. 1. Classification accuracy on 20 Newsgroups (top), WebKB (middle) and 7 Sectors (bottom). Curves show small error bars twice the width of the standard error of the mean. Differences between WAPMI and the other metrics are statistically significant (at the 95% confidence level using a two-tailed paired t-test) at the following vocabulary sizes: on 20 Newsgroups from 20 to 50,000 words; on WebKB from 20 to 2,000 words; on 7 Sectors from 100 to 20,000 words.

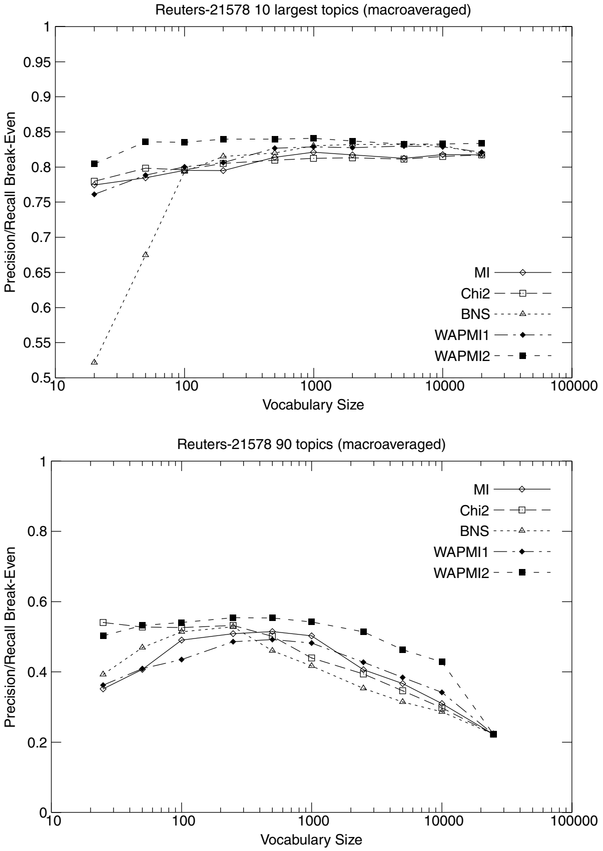

Fig. 2. Macroaveraged precision/recall breakeven on the Reuters datasets with 10 (top) and 90 (bottom) topic classes. WAPMI1 gives equal weight to documents, while WAPMI2 gives equal weight to classes.

Table 3 shows the results. For each dataset and each scoring function we report the number of features and the classification performance at the selected threshold. In addition we show the classification performance at the full vocabulary (i.e. with no feature selection).

We make two observations in Table 3. First, WAPMI is always among the top performers, although its performance is significantly better only on 20 Newsgroups and Reuters. Mutual Information performs significantly worse than the other metrics on 7 Sectors. Secondly and more importantly, the number of features selected by WAPMI seems to reflect the difficulty of the datasets better than for the other scoring methods. For 20 Newsgroups, which requires many features, WAPMI1 selects more features than any other method, while it still omits some features which results in an improvement of 2 percentage points compared to the full vocabulary. In contrast, the WAPMI scores select considerably less features on the Reuters datasets than the other methods, with better results.

Table 3. Global thresholding results. Shown are the number of selected words at the predefined threshold, classification performance, and standard deviation where applicable. Statistically significant differences (at $p = 0.95$ using a two-tailed paired t-test) are printed in boldface. For Reuters, macroaveraged precision/recall breakeven points are shown.

	20 Newsgroups			WebKB			7 Sectors		
	Words	Acc	SDev	Words	Acc	SDev	Words	Acc	SDev
Chi^2=0.1	65,194	81.35%	0.36%	32,712	84.79%	0.99%	15,147	72.29%	1.19%
MI=10^{-7}	77,694	81.13%	0.37%	32,776	84.79%	1.01%	37,474	**68.32%**	1.17%
BNS=0.05	62,777	81.42%	0.26%	32,550	84.78%	0.99%	8,545	72.27%	1.78%
WAPMI1=0	85,870	**82.92%**	0.72%	32,091	85.00%	0.96%	37,422	73.12%	0.57%
WAPMI2=0				32,278	85.06%	1.03%	37,428	73.14%	1.01%
Full	86,019	80.97%	0.29%	32,873	84.80%	0.99%	37,474	68.32%	1.17%

	Reuters-10		Reuters-90	
	Words	P/R	Words	P/R
Chi^2=0.1	18,861	81.72%	23,395	22.30%
MI=10^{-7}	18,014	81.72%	22,571	22.57%
BNS=0.05	20,086	81.76%	23,778	22.26%
WAPMI1=0	7,617	82.47%	3,066	**44.58%**
WAPMI2=0	10,610	**83.17%**	20,762	38.97%
Full	22,430	81.61%	24,719	22.28%

5 Conclusions

This paper proposes weighted average pointwise mutual information (WAPMI) as a replacement for mutual information to rank features for feature selection in text categorization. Experiments on a number of standard benchmark datasets show that WAPMI outperforms several other feature scoring metrics, including mutual information, Chi-squared and Bi-normal separation. An important property of WAPMI is that the feature set size (i.e. the number of selected features) can be set automatically, depending on the complexity and difficulty of the dataset, by using a simple constant-threshold heuristics that maximizes an objective function and does not require EM or model selection.

WAPMI contains weights that can be set to account for skewed class distributions, which we used in our experiments with the Reuters dataset and obtained improved classification performance. It is not entirely clear how this could be done with other metrics.

We have used WAPMI with the multinomial Naive Bayes classifier, but future work should deal with other classification models, e.g. support vector machines. A general open problem is that feature selection for multinomial Naive Bayes is not entirely well-defined, thus we are actually approximating feature selection. More work is required to better understand how feature selection affects the class-conditional distributions.

Acknowledgments

The author would like to thank the anonymous reviewers for their detailed comments and suggestions that helped to improve the paper.

References

1. John, G.H., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In Cohen, W.W., Hirsh, H., eds.: Machine Learning: Proceedings of the Eleventh International Conference, San Francisco, CA, Morgan Kaufmann Publishers (1994) 121–129
2. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Proc. 14th International Conference on Machine Learning (ICML-97). (1997) 412–420
3. McCallum, A., Nigam, K.: A comparison of event models for Naive Bayes text classification. In: Learning for Text Categorization: Papers from the AAAI Workshop, AAAI Press (1998) 41–48 Technical Report WS-98-05.
4. Eyheramendy, S., Lewis, D.D., Madigan, D.: On the Naive Bayes model for text categorization. In Bishop, C.M., Frey, B.J., eds.: AI & Statistics 2003: Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics. (2003) 332–339
5. Friedman, J.H.: On bias, variance, 0/1-loss, and the curse-of-dimensionality. Data Mining and Knowledge Discovery **1** (1997) 55–77
6. Cover, T.M., Thomas, J.A.: Elements of Information Theory. John Wiley, New York (1991)
7. Church, K.W., Hanks, P.: Word association norms, mutual information, and lexicography. Computational Linguistics **16** (1990) 22–29
8. Rennie, J.D.M.: Improving multi-class text classification with Naive Bayes. Master's thesis, Massachusetts Institute of Technology (2001)
9. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM **18** (1975) 613–620
10. Dhillon, I.S., Mallela, S., Kumar, R.: A divisive information-theoretic feature clustering algorithm for text classification. Journal of Machine Learning Research **3** (2003) 1265–1287
11. Lang, K.: NewsWeeder: Learning to filter netnews. In: Proc. 12th International Conference on Machine Learning (ICML-95), Morgan Kaufmann (1995) 331–339
12. Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S.: Learning to construct knowledge bases from the World Wide Web. Artificial Intelligence **118** (2000) 69–113
13. Apté, C., Damerau, F., Weiss, S.M.: Towards language independent automated learning of text categorization models. In: Proc. 17th ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '94). (1994) 23–30
14. Forman, G.: An extensive empirical study of feature selection metrics for text classification. Journal of Machine Learning Research **3** (2003) 1289–1305