A Comparison of Event Models for Naive Bayes Anti-Spam E-Mail Filtering

Karl-Michael Schneider
University of Passau
Department of General Linguistics
Innstr. 40, D-94032 Passau
schneide@phil.uni-passau.de

Abstract

We describe experiments with a Naive Bayes text classifier in the context of anti-spam E-mail filtering, using two different statistical event models: a multi-variate Bernoulli model and a multinomial model. We introduce a family of feature ranking functions for feature selection in the multinomial event model that take account of the word frequency information. We present evaluation results on two publicly available corpora of legitimate and spam E-mails. We find that the multinomial model is less biased towards one class and achieves slightly higher accuracy than the multi-variate Bernoulli model.

1 Introduction

Text categorization is the task of assigning a text document to one of several predefined categories. Text categorization plays an important role in natural language processing (NLP) and information retrieval (IR) applications. One particular application of text categorization is anti-spam E-mail filtering, where the goal is to block unsolicited messages with commercial or pornographic content (UCE, spam) from a user's E-mail stream, while letting other (legitimate) messages pass. Here, the task is to assign a message to one of two categories, *legitimate* and *spam*, based on the message's content.

In recent years, a growing body of research has applied machine learning techniques to text categorization and (anti-spam) E-mail filtering, including rule learning (Cohen, 1996), Naive Bayes (Sahami et al., 1998; Androutsopoulos et al., 2000b; Rennie, 2000), memory based learning (Androutsopoulos et al., 2000b), decision trees (Carreras and Màrquez, 2001), support vector machines (Drucker et al., 1999) or combinations of different learners (Sakkis et al., 2001). In these approaches a classifier is learned from training data rather than constructed by hand, which results in better and more robust classifiers.

The Naive Bayes classifier has been found particularly attractive for the task of text categorization because it performs surprisingly well in many application areas despite its simplicity (Lewis, 1998). Bayesian classifiers are based on a probabilistic model of text generation. A text is generated by first choosing a class according to some prior probability and then generating a text according to a class-specific distribution. The model parameters are estimated from training examples that have been annotated with their correct class. Given a new document, the classifier outputs the class which is most likely to have generated the document.

From a linguistic point of view, a document is made up of words, and the semantics of the document is determined by the meaning of the words and the linguistic structure of the document. The Naive Bayesian classifier makes the simplifying assumption that the probability that a document is generated in some class depends only on the probabilities of the words given the context of the class, and that the words in a document are independent of each other. This is called the *Naive Bayes assumption.*

The generative model underlying the Naive Bayes classifier can be characterized with respect to the amount of information it captures about the

words in a document. In information retrieval and text categorization, two types of models have been used (McCallum and Nigam, 1998). Both assume that there is a fixed vocabulary. In the first model, a document is generated by first choosing a subset of the vocabulary and then using the selected words any number of times, at least once, in any order. This model is called *multi-variate Bernoulli model*. It captures the information of which words are used in a document, but not the number of times each words is used, nor the order of the words in the document.

In the second model, a document is generated by choosing a set of word occurrences and arranging them in any order. This model is called *multinomial model*. In addition to the multi-variate Bernoulli model, it also captures the information about how many times a word is used in a document. Note that in both models, a document can contain additional words that are not in the vocabulary, which are considered noise and are not used for classification.

Despite the fact that the multi-variate Bernoulli model captures less information about a document (compared to the multinomial model), it performs quite well in text categorization tasks, particularly when the set of words used for classification is small. However, McCallum and Nigam (1998) have shown that the multinomial model outperforms the multi-variate Bernoulli model on larger vocabulary sizes or when the vocabulary size is chosen optimal for both models.

Most text categorization approaches to anti-spam E-mail filtering have used the multi-variate Bernoulli model (Androutsopoulos et al., 2000b). Rennie (2000) used a multinomial model but did not compare it to the multi-variate model. Mladenić and Grobelnik (1999) used a multinomial model in a different context. In this paper we present results of experiments in which we evaluated the performance of a Naive Bayes classifier on two publicly available E-mail corpora, using both the multi-variate Bernoulli and the multinomial model.

The paper is organized as follows. In Sect. 2 we describe the Naive Bayes classifier and the two generative models in more detail. In Sect. 3 we introduce feature selection methods that take into account the extra information contained in the multinomial model. In Sect. 4 we describe our experiments and discuss the results. Finally, in Sect. 5 we draw some conclusions.

2 Naive Bayes Classifier

We follow the description of the Naive Bayes classifier given in McCallum and Nigam (1998). A Bayesian classifier assumes that a document is generated by a mixture model with parameters θ, consisting of components $\mathcal{C} = \{c_1, \ldots, c_n\}$ that correspond to the classes. A document is generated by first selecting a component $c_j \in \mathcal{C}$ according to the prior distribution $P(c_j|\theta)$ and then choosing a document d_i according to the parameters of c_j with distribution $P(d_i|c_j;\theta)$. The likelihood of a document is given by the total probability

$$P(d_i|\theta) = \sum_{j=1}^{|\mathcal{C}|} P(c_j|\theta)P(d_i|c_j;\theta) \qquad (1)$$

Of course, the true parameters θ of the mixture model are not known. Therefore, one estimates the parameters from labeled training documents, i.e. documents that have been manually annotated with their correct class. We denote the estimated parameters with $\hat{\theta}$. Given a set of training documents $\mathcal{D} = \{d_1, \ldots, d_m\}$, the class prior parameters are estimated as the fraction of training documents in c_j, using maximum likelihood:

$$\hat{\theta}_{c_j} = P(c_j|\hat{\theta}) = \frac{\sum_{i=1}^{|\mathcal{D}|} P(c_j|d_i)}{|\mathcal{D}|} \qquad (2)$$

where $P(c_j|d_i)$ is 1 if $d_i \in c_j$ and 0 otherwise. The estimation of $P(d_i|c_j;\theta)$ depends on the generative model and is described below.

Given a new (unseen) document d, classification of d is performed by computing the posterior probability of each class, given d, by applying Bayes' rule:

$$P(c_j|d;\hat{\theta}) = \frac{P(c_j|\hat{\theta})P(d|c_j;\hat{\theta})}{P(d|\hat{\theta})} \qquad (3)$$

The classifier simply selects the class with the highest posterior probability. Note that $P(d|\hat{\theta})$ is

the same for all classes, thus d can be classified by computing

$$c_d = \underset{c_j \in \mathcal{C}}{\operatorname{argmax}} P(c_j|\hat{\theta}) P(d|c_j;\hat{\theta}) \quad (4)$$

2.1 Multi-variate Bernoulli Model

The multi-variate Bernoulli event model assumes that a document is generated by a series of $|V|$ Bernoulli experiments, one for each word w_t in the vocabulary V. The outcome of each experiment determines whether the corresponding word will be included at least once in the document. Thus a document d_i can be represented as a binary feature vector of length $|V|$, where each dimension t of the vector, denoted as $B_{it} \in \{0, 1\}$, indicates whether word w_t occurs at least once in d_i. The Naive Bayes assumption assumes that the $|V|$ trials are independent of each other. By making the Naive Bayes assumption, we can compute the probability of a document given a class from the probabilities of the words given the class:

$$P(d_i|c_j;\theta) = \prod_{t=1}^{|V|} (B_{it} P(w_t|c_j;\theta) + (1 - B_{it})(1 - P(w_t|c_j;\theta))) \quad (5)$$

Note that words which do not occur in d_i contribute to the probability of d_i as well. The parameters $\theta_{w_t|c_j} = P(w_t|c_j;\theta)$ of the mixture component c_j can be estimated as the fraction of training documents in c_j that contain w_t:[1]

$$\hat{\theta}_{w_t|c_j} = P(w_t|c_j;\hat{\theta}) = \frac{\sum_{i=1}^{|\mathcal{D}|} B_{it} P(c_j|d_i)}{\sum_{i=1}^{|\mathcal{D}|} P(c_j|d_i)} \quad (6)$$

2.2 Multinomial Model

The multinomial event model assumes that a document d_i of length $|d_i|$ is generated by a sequence of $|d_i|$ word events, where the outcome of each event is a word from the vocabulary V. Following McCallum and Nigam (1998), we assume that the document length distribution $P(|d_i|)$ does not depend on the class. Thus a document d_i can be represented as a vector of length $|V|$, where each dimension t of the vector, denoted as $N_{it} \geq 0$, is the number of times word w_t occurs in d_i. The Naive Bayes assumption assumes that the $|d_i|$ trials are independent of each other. By making the Naive Bayes assumption, the probability of a document given a class is the multinomial distribution:

$$P(d_i|c_j;\theta) = P(|d_i|)|d_i|! \prod_{t=1}^{|V|} \frac{P(w_t|c_j;\theta)^{N_{it}}}{N_{it}!} \quad (7)$$

The parameters $\theta_{w_t|c_j} = P(w_t|c_j;\theta)$ of the mixture component c_j can be estimated as the fraction of word occurrences in the training documents in c_j that are w_t:

$$\hat{\theta}_{w_t|c_j} = P(w_t|c_j;\hat{\theta}) = \frac{\sum_{i=1}^{|\mathcal{D}|} N_{it} P(c_j|d_i)}{\sum_{s=1}^{|V|} \sum_{i=1}^{|\mathcal{D}|} N_{is} P(c_j|d_i)} \quad (8)$$

3 Feature Selection

3.1 Mutual Information

It is common to use only a subset of the vocabulary for classification, in order to reduce over-fitting to the training data and to speed up the classification process. Following McCallum and Nigam (1998) and Androutsopoulos et al. (2000b), we ranked the words according to their average mutual information with the class variable and selected the N highest ranked words. Average mutual information between a word w_t and the class variable, denoted by $MI(C; W_t)$, is the difference between the entropy of the class variable, $H(C)$, and the entropy of the class variable given the information about the word, $H(C|W_t)$ (Cover and Thomas, 1991). Intuitively, $MI(C; W_t)$ measures how much bandwidth can be saved in the transmission of a class value when the information about the word is known.

In the multi-variate Bernoulli model, W_t is a random variable that takes on values $f_t \in \{0, 1\}$, indicating whether word w_t occurs in a document or not. Thus $MI(C; W_t)$ is the average mutual information between C and the absence or presence of w_t in a document:

$$MI(C; W_t) = \sum_{c \in C} \sum_{f_t \in \{0,1\}} P(c, f_t) \log \frac{P(c, f_t)}{P(c) P(f_t)} \quad (9)$$

[1] McCallum and Nigam (1998) suggest to use a Laplacean prior to smooth the probabilities, but we found that this degraded the performance of the classifier.

3.2 Feature Selection in the Multinomial Model

Mutual information as in (9) has also been used for feature selection in the multinomial model, either by estimating the probabilities $P(c, f_t)$, $P(c)$ and $P(f_t)$ as in the multi-variate model (McCallum and Nigam, 1998) or by using the multinomial probabilities (Mladenić and Grobelnik, 1999). Let us call the two versions *mv-MI* and *mn-MI*, respectively.

mn-MI is not fully adequate as a feature ranking function for the multinomial model. For example, the token `Subject:` appears in every document in the two corpora we used in our experiments exactly once, and thus is completely uninformative. *mv-MI* assigns 0 to this token, but *mn-MI* yields a positive value because the average document length in the classes is different, and thus the class-conditional probabilities of the token are different across classes in the multinomial model. On the other hand, assume that some token occurs once in every document in c_1 and twice in every document in c_2, and that the average document length in c_2 is twice the average document length in c_1. Then both *mv-MI* and *mn-MI* will assign 0 to the token, although it is clearly highly informative in the multinomial model.

We experimented with feature scoring functions that take into account the average number of times a word occurs in a document. Let $N(c_j, w_t) = \sum_{i=1}^{|\mathcal{D}|} N_{it} P(c_j|d_i)$, $N(c_j) = \sum_{t=1}^{|V|} N(c_j, w_l)$ and $N(w_t) = \sum_{j=1}^{|\mathcal{C}|} N(c_j, w_t)$ denote the number of times word w_t occurs in class c_j, the total number of word occurrences in c_j, and the total number of occurrences of w_t, respectively. Let $d(c_j) = \sum_{i=1}^{|\mathcal{D}|} P(c_j|d_i)$ denote the number of documents in c_j. Then the average number of times w_t occurs in a document in c_j is defined by $mtf(c_j, w_t) = \frac{N(c_j,w_t)}{d(c_j)}$ (*mean term frequency*). The average number of times w_t occurs in a document is defined by $mtf(w_t) = \frac{N(w_t)}{|D|}$.

In the multinomial model, a word is informative with respect to the class value if its mean term frequency in some class is different from its (global) mean term frequency, i.e. if $\frac{mtf(c_j,w_t)}{mtf(w_t)} \neq 1$. We used feature ranking functions of the form in (10):

$$I_{f,R}(C; w_t) = \sum_{j=1}^{|\mathcal{C}|} f(c_j, w_t) \log R(c_j, w_t) \quad (10)$$

$R(c_j, w_l)$ measures the amount of information that w_t gives about c_j. $f(c_j, w_t)$ is a weighting function. Table 1 lists the feature ranking functions that we used in our experiments. *mn-MI* is the average mutual information where the probabilities are estimated as in the multinomial model. *dmn-MI* differs from *mn-MI* in that the class prior probabilities are estimated as the fraction of documents in each class, rather than the fraction of word occurrences. *tf-MI*, *dtf-MI* and *tftf-MI* use mean term frequency to measure the correlation between w_t and c_j and use different weighting functions.

4 Experiments

4.1 Corpora

We performed experiments on two publicly available E-mail corpora:[2] *Ling-Spam* (Androutsopoulos et al., 2000b) and *PU1* (Androutsopoulos et al., 2000a). We trained a Naive Bayes classifier with a multi-variate Bernoulli model and a multinomial model on each of the two datasets.

The *Ling-Spam* corpus consists of 2412 messages from the Linguist list[3] and 481 spam messages. Thus spam messages are 16.6% of the corpus. Attachments, HTML tags and all E-mail headers except the Subject line have been stripped off. We used the lemmatized version of the corpus, with the tokenization given in the corpus and with no additional processing, stop list, etc. The total vocabulary size is 59829 words.

The *PU1* corpus consists of 618 English legitimate messages and 481 spam messages. Messages in this corpus are encrypted: Each token has been replaced by a unique number, such that different occurrences of the same token get the same number (the only non encrypted token is the `Subject:` header name). Spam messages are 43.8% of the corpus. As with the *Ling-Spam* corpus, we used the lemmatized version with no ad-

[2] available from the publications section of `http://www.aueb.gr/users/ion/`

[3] `http://www.linguistlist.org/`

Name	$f(c_j, w_t)$	$R(c_j, w_t)$
mn-MI	$P(c_j, w_t) = \frac{N(c_j, w_t)}{\sum_{s=1}^{\|V\|} N(w_s)}$	$\frac{P(c_j, w_t)}{P(c_j)P(w_t)} = \frac{N(c_j, w_t)}{N(c_j)} \cdot \frac{\sum_{s=1}^{\|V\|} N(w_s)}{N(w_t)}$
dmn-MI	$P(w_t\|c_j)P(c_j) = \frac{N(c_j, w_t)}{N(c_j)} \cdot \frac{d(c_j)}{\|\mathcal{D}\|}$	$\frac{P(w_t\|c_j)}{P(w_t)} = \frac{N(c_j, w_t)/N(c_j)}{\sum_{k=1}^{\|\mathcal{C}\|} P(c_k)(N(c_k, w_t)/N(c_k))}$
tf-MI	$P(c_j, w_t) = \frac{N(c_j, w_t)}{\sum_{s=1}^{\|V\|} N(w_s)}$	$\frac{mtf(c_j, w_t)}{mtf(w_t)} = \frac{N(c_j, w_t)}{d(c_j)} \cdot \frac{\|\mathcal{D}\|}{N(w_t)}$
dtf-MI	$P(w_t\|c_j)P(c_j) = \frac{N(c_j, w_l)}{N(c_j)} \cdot \frac{d(c_j)}{\|\mathcal{D}\|}$	$\frac{mtf(c_j, w_l)}{mtf(w_t)} = \frac{N(c_j, w_l)}{d(c_j)} \cdot \frac{\|\mathcal{D}\|}{N(w_t)}$
tftf-MI	$mtf(c_j, w_t) = \frac{N(c_j, w_t)}{d(c_j)}$	$\frac{mtf(c_j, w_t)}{mtf(w_t)} = \frac{N(c_j, w_t)}{d(c_j)} \cdot \frac{\|\mathcal{D}\|}{N(w_t)}$

Table 1: Feature ranking functions for the multinomial event model (see text).

ditional processing. The total vocabulary size is 21706 words.

Both corpora are divided into 10 parts of equal size, with equal proportion of legitimate and spam messages across the 10 parts. Following (Androutsopoulos et al., 2000b), we used 10-fold cross-validation in all experiments, using nine parts for training and the remaining part for testing, with a different test set in each trial. The evaluation measures were then averaged across the 10 iterations.

We performed experiments on each of the corpora, using the multi-variate Bernoulli model with *mv-MI*, as well as the multinomial model with *mv-MI* and the feature ranking functions in Table 1, and varying the number of selected words from 50 to 5000 by 50.

4.2 Results

For each event model and feature ranking function, we determined the minimum number of words with highest recall for which recall equaled precision (*breakeven point*). Tables 2 and 3 present the breakeven points with the number of selected words, recall in each class, and accuracy. In some cases, precision and recall were different over the entire range of the number of selected words. In these cases we give the recall and accuracy for the minimum number of words for which accuracy was highest.

Figures 1 and 2 show recall curves for the multi-variate Bernoulli model and three feature ranking functions in the multinomial model for *Ling-Spam*, and Figures 3 and 4 for *PU1*.

Some observations can be made from these results. First, the multi-variate Bernoulli model favors the *Ling* resp. *Legit* classes over the *Spam* classes, whereas the multinomial model is more balanced in conjunction with *mv-MI*, *tf-MI* and *tftf-MI*. This may be due to the relatively specific vocabulary used especially in the *Ling-Spam* corpus, and to the uneven distribution of the documents in the classes. Second, the multinomial model achieves higher accuracy than the multi-variate Bernoulli model. *tf-MI* even achieves high accuracy at a comparatively small vocabulary size (1200 and 2400 words, respectively). In general, *PU1* seems to be more difficult to classify.

Androutsopoulos et al. (2000b) used cost-sensitive evaluation metrics to account for the fact that it may be more serious an error when a legitimate message is classified as spam than vice versa. However, such cost-sensitive measures are problematic with a Naive Bayes classifier because the probabilities computed by Naive Bayes are not reliable, due to the independence assumptions it makes. Therefore we did not use cost-sensitive measures.[4]

5 Conclusions

We performed experiments with two different statistical event models (a multi-variate Bernoulli

[4]Despite this, Naive Bayes can be an optimal classifier because it uses only the ranking implied by the probabilities, not the probabilities themselves (Domingos and Pazzani, 1997).

Name	Vocabulary size	*Ling*	*Spam*	Accuracy
Bernoulli	*3900*	***99.88***	*88.57*	*98.00*
mv-MI	1050	99.34	**96.47**	**98.86**
mn-MI	200	98.92	93.76	98.06
dmn-MI	*500*	*99.21*	*16.84*	*85.52*
tf-MI	1200	99.34	**96.05**	**98.79**
dtf-MI	200	99.09	95.43	**98.48**
tftf-MI	4550	99.30	**96.26**	**98.79**

Table 2: Precision/recall breakeven points for *Ling-Spam*. Rows printed in italic show the point of maximum accuracy in cases where precision and recall were different for all vocabulary sizes. Values that are no more than 0.5% below the highest value in a column are printed in bold.

Name	Vocabulary size	*Legit*	*Spam*	Accuracy
Bernoulli	*4800*	*98.54*	*92.52*	*95.91*
mv-MI	4450	97.41	96.67	**97.09**
mn-MI	900	96.44	95.43	96.00
dmn-MI	*4400*	***99.51***	*92.52*	*96.45*
tf-MI	2400	97.73	**97.09**	**97.45**
dtf-MI	1000	96.60	95.63	96.18
tftf-MI	*2600*	*97.57*	***97.30***	***97.45***

Table 3: Precision/recall breakeven points for *PU1*.

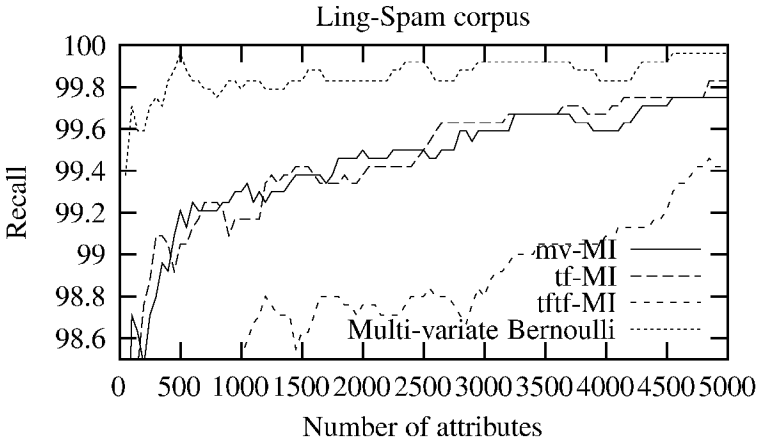

Figure 1: *Ling* recall in the *Ling-Spam* corpus for different feature ranking functions and at different vocabulary sizes. *mv-MI*, *tf-MI* and *tftf-MI* use the multinomial event model.

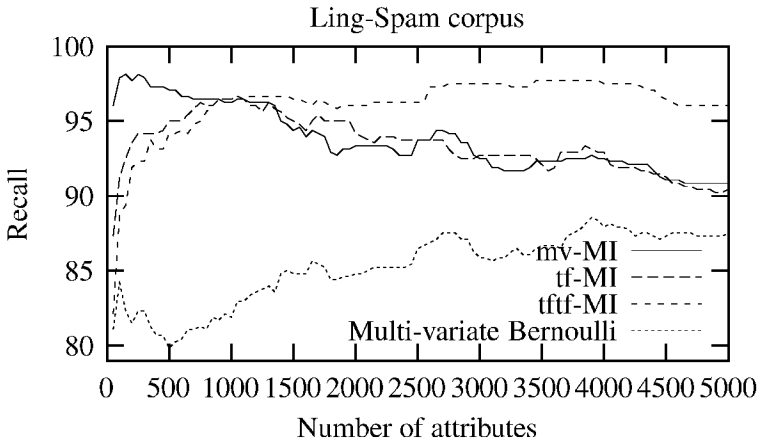

Figure 2: *Spam* recall in the *Ling-Spam* corpus at different vocabulary sizes.

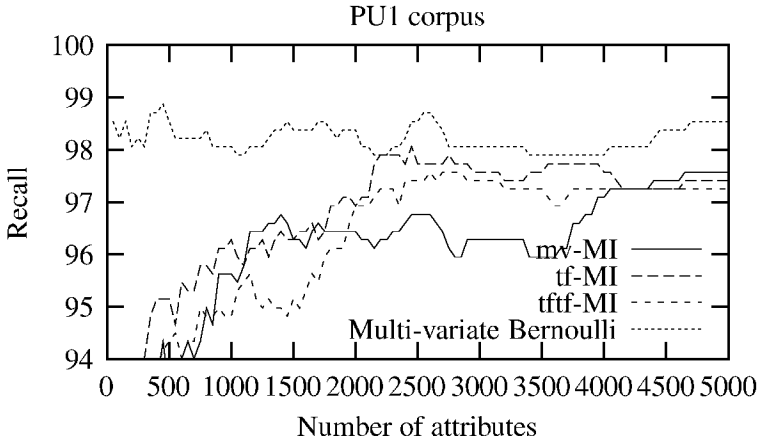

Figure 3: *Legitimate* recall in the *PU1* corpus at different vocabulary sizes.

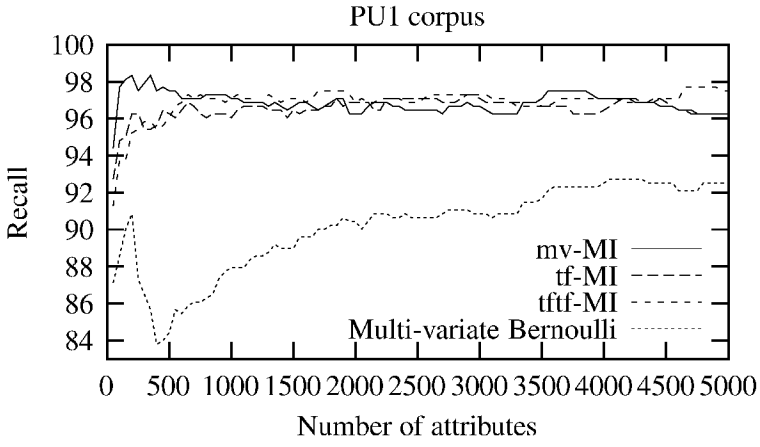

Figure 4: *Spam* recall in the *PU1* corpus at different vocabulary sizes.

model and a multinomial model) for a Naive Bayes text classifier using two publicly available E-mail corpora. We used several feature ranking functions for feature selection in the multinomial model that explicitly take into account the word frequency information contained in the multinomial document representation. The main conclusion we draw from these experiments is that the multinomial model is less biased towards one class and can achieve higher accuracy than the multivariate Bernoulli model, in particular when frequency information is taken into account also in the feature selection process.

Our plans for future work are to evaluate the feature selection functions for the multinomial model introduced in this paper on other corpora, and to provide a better theoretical foundation for these functions. Most studies on feature selection have concentrated on the multi-variate Bernoulli model (Yang and Pedersen, 1997). We believe that the information contained in the multinomial document representation has been neglected in previous studies, and that the development of feature selection functions especially for the multinomial model could improve its performance.

References

Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, and Constantine D. Spyropoulos. 2000a. An experimental comparison of Naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In N. J. Belkin, P. Inwersen, and M.-K. Leong, editors, *Proc. 23rd ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2000)*, pages 160–167, Athens, Greece.

Ion Androutsopoulos, Georgios Paliouras, Vangelis Karkaletsis, Georgios Sakkis, Constantine D. Spyropoulos, and Panagiotis Stamatopoulos. 2000b. Learning to filter spam e-mail: A comparison of a Naive Bayesian and a memory-based approach. In H. Zaragoza, P. Gallinari, and M. Rajman, editors, *Proc. Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2000)*, pages 1–13, Lyon, France.

Xavier Carreras and Lluís Màrquez. 2001. Boosting trees for anti-spam email filtering. In *Proc. International Conference on Recent Advances in Natural Language Processing (RANLP-01)*, Tzigov Chark, Bulgaria.

William W. Cohen. 1996. Learning rules that classify e-mail. In *Papers from the AAAI Spring Symposium on Machine Learning in Information Access*, pages 18–25, Stanford, CA. AAAI Press.

Thomas M. Cover and Joy A. Thomas. 1991. *Elements of Information Theory*. John Wiley, New York.

Pedro Domingos and Michael Pazzani. 1997. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130.

Harris Drucker, Donghui Wu, and Vladimir N. Vapnik. 1999. Support vector machines for spam categorization. *IEEE Trans. on Neural Networks*, 10(5):1048–1054.

David D. Lewis. 1998. Naive (Bayes) at forty: The independence assumption in information retrieval. In *Proc. 10th European Conference on Machine Learning (ECML98)*, volume 1398 of *Lecture Notes in Computer Science*, pages 4–15, Heidelberg. Springer.

Andrew McCallum and Kamal Nigam. 1998. A comparison of event models for Naive Bayes text classification. In *Proc. AAAI-98 Workshop on Learning for Text Categorization*, pages 41–48. AAAI Press.

Dunja Mladenić and Marko Grobelnik. 1999. Feature selection for unbalanced class distribution and Naive Bayes. In I. Bratko and S. Dzeroski, editors, *Proc. 16th International Conference on Machine Learning (ICML-99)*, pages 258–267, San Francisco, CA. Morgan Kaufmann Publishers.

Jason D. M. Rennie. 2000. ifile: An application of machine learning to e-mail filtering. In *Proc. KDD-2000 Workshop on Text Mining*, Boston, MA.

Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. 1998. A bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the AAAI Workshop*, pages 55–62, Madison Wisconsin. AAAI Press. Technical Report WS-98-05.

Georgios Sakkis, Ion Androutsopoulos, Georgios Paliouras, Vangelis Karkaletsis, Constantine D. Spyropoulos, and Panagiotis Stamatopoulos. 2001. Stacking classifiers for anti-spam filtering of e-mail. In L. Lee and D. Harman, editors, *Proc. 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, pages 44–50, Pittsburgh, PA. Carnegie Mellon University.

Yiming Yang and Jan O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Proc. 14th International Conference on Machine Learning (ICML-97)*, pages 412–420.