Feature Selection with Conditional Mutual Information MaxiMin in Text Categorization

Gang Wang
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
wanggang@cs.ust.hk

Frederick H. Lochovsky
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
fred@cs.ust.hk

Qiang Yang
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
qyang@cs.ust.hk

ABSTRACT

Feature selection is an important component of text categorization. This technique can both increase a classifier's computation speed, and reduce the overfitting problem. Several feature selection methods, such as information gain and mutual information, have been widely used. Although they greatly improve the classifier's performance, they have a common drawback, which is that they do not consider the mutual relationships among the features. In this situation, where one feature's predictive power is weakened by others, and where the selected features tend to bias towards major categories, such selection methods are not very effective. In this paper, we propose a novel feature selection method for text categorization called *conditional mutual information maximin* (CMIM). It can select a set of individually discriminating and weakly dependent features. The experimental results show that CMIM can perform much better than traditional feature selection methods.

Categories and Subject Descriptors

H.4.m [**Information Systems Applications**]: Miscellaneous; I.2.6 [**Artificial Intelligence**]: Learning—*Induction*; I.5.4 [**Pattern Recognition**]: Application—*Text Processing*

General Terms

Theory, Algorithms, Performance

Keywords

Conditional mutual information, feature selection, classification, text categorization, Naïve Bayes, SVMs

CIKM'04, November 8–13, 2004, Washington, DC, USA.
Copyright 2004 ACM 1-58113-874-1/04/0011 ...$5.00.

1. INTRODUCTION

Due to the increased availability of documents in digital formats, there is a growing need for better methods to find, filter, and manage these resources. Text categorization - the assignment of natural language texts to one or more predefined categories based on their content - is a major solution to these kinds of problems. It has several applications such as assigning subject categories to documents to support text retrieval, routing and filtering. In recent years, inductive learning algorithms have shown great benefit over rule-based approaches in text categorization, because they do not need time-consuming and costly manual work. The most popular inductive learning algorithms include LLSF [23], decision tree, naïve bayes [12, 14], kNN, SVMs [20, 10] and maximum entropy methods [15].

The major difficulty in text categorization problems is caused by the high dimensionality of the feature space. Since most inductive learning algorithms are based on error minimization and numerical optimization, reducing the number of features can not only increase the efficiency of the classifier, but also reduce the overfitting problems. Automatic dimension reduction methods include the removal of non-informative terms according to corpus statistics, or the construction of new features; i.e. combining lower-level features into higher-level orthogonal dimensions. Many methods have been investigated, and generally they are classified into two categories in terms of the nature of the resulting terms: term selection and term extraction.

For *term selection*, the elements in the selected feature set are original terms in a corpus. Terms are ranked according to their individual predictive power, which is estimated by various methods such as document frequency, information gain, mutual information [21, 24], or χ^2-test [7]. The underlying intuition is that the best terms are the ones that are distributed most differently in the sets of positive and negative examples in different categories. Many researchers [2, 9, 13, 24] have comparatively studied different term selection methods, and they showed that feature selection could result in a moderate increase in effectiveness compared with no feature selection. Although each term selection function has its own rationale, they indeed have a common drawback. That is, the value for each selected term is only the benefit it brings individually, regardless of the overall importance measurement for all selected terms. Because the top ranked terms are always correlated with each other, the information a term set brings is usually much less than the summation of the information brought by each term in the set. This circumstance will be even worse when some categories do not have enough training examples. It is difficult to find their representative features since delineative features for rare categories have less ranking value than those non-representative features in major categories. Thus, the features chosen by traditional feature selection methods will always be biased towards some major categories. A sound feature selection strategy is to discovery the Markov Blanket in the faithful Bayesian network [19]. But due to its inefficiency for high dimensional dataset, it's seldom used. Looking for features as orthogonal as possible, such as is done in the PCA method, is another solution to this problem.

Term extraction is another approach to deal with dimension reduction. It can effectively deal with the problem mentioned above [18]. With this approach, the selected features are not original terms in a corpus, but synthetic terms that maximize the overall effectiveness. Latent Semantic Indexing (LSI) [5, 1] is such a technique that compresses document vectors into vectors in a lower-dimensional space whose dimensions are obtained as combinations of the original dimensions by looking at their patterns of occurrence. LSI in fact infers the dependence among the original terms from documents and "wires" this dependence into the newly obtained, independent dimensions. It uses a matrix decomposition technique - singular value decomposition - to form a new vector space. This subspace represents important associative relationships between terms and documents that are not evident in individual documents. The features not only can be ranked according to their importance, but also thus are independent of each other. LSI is an effective approach for dimension reduction, and has been applied with remarkable success[17]. But it has a number of shortcomings as well, mainly due to its complexity, and unsatisfactory statistical foundation. Factor analysis (FA) and independent component analysis (ICA) also belong to this approach, but they are difficult to apply to real-life large scale, dynamic situations.

To tackle the shortcomings of the above two methods for dimension reduction, we propose a tradeoff algorithm between redundancy and discrimination. In our approach, the selected features are original terms in a corpus similar to that for feature selection. Our approach also ensures a good orthogonality between features like feature extraction. In this way it combines the advantages of both methods. It can select individually discriminating and weakly dependent features, which ensures that the feature set will not be biased towards some major categories. The fundamental idea of our algorithm is that it selects a term that maximizes its mutual information with the class to predict, and does this conditionally in response to features already picked up in the feature set. This conditional mutual information maximin (CMIM) criterion will not select features correlated to ones already picked, even if they are powerful individually, since they do not carry any additional information for the classification.

The organization of this paper is as follows. Section 2 introduces the CMIM algorithm and related work. Two classifiers used in our experiments will be briefly reviewed in section 3. Section 4 reports our experiment setup and result, where we compare our feature selection methods with others. The last section summarizes the concluding remarks and the future work.

2. CONDITIONAL MUTUAL INFORMATION MAXIMIN

2.1 Information Theory Review

Information theory has been widely used in many aspects, such as science, engineering, business, etc [3].It is an umbrella term for the scientific disciplines that attempt to codify the mathematical underpinnings of data. That is why many feature selection methods, such as information gain, mutual information, are based on this concept. In our discussion we only deal with discrete values, and we denote the discrete random variables as X and Y.

A quantity called the entropy, $H(X)$, is the most fundamental concept in information theory, since it has many properties that agree with the intuitive notion of what a measure of information should be. $H(X)$ is defined by

$$H(X) = -\sum_{x \in X} p(x) \log p(x)$$

Note that entropy does not depend on the actual values taken by the random variable X; it depends only on the probabilities. We will take all logarithms to base 2. Hence the value of the entropy quantifies how many bits are required to encode or describe the random variable X on average. The entropy is at its maximum when a random variable has a uniform distribution on its values.

The mutual information (MI), $I(X,Y)$, defined as

$$I(X;Y) = -\sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

quantifies how much information is shared between X and Y. MI has been widely used as a feature selection method. $I(C;F_i)$ can measure the goodness of a term globally between the category variable C and a feature F_i (We only consider 1-of-n classification problem, i.e., assigning each document into only one category in this paper. The cardinality of C is the number of all categories). Those features with high predictive power have larger MI values. If C and F_i are independent, their MI has a minimum value of zero. Thus, we will select features whose mutual information with category are the largest.

The main goal of feature selection is to select a small number of features that can carry as much information as possible. This goal can be interpreted as maximizing the joint MI (JMI), $I(F_1, \ldots, F_k; C)$, for k features. [22] tried to estimate JMI by directly estimating the joint probability $p(f_1, \ldots, f_k, c)$. However, this method suffers from the curse of dimensionality when k is large. Assume that all random variables involved are discrete, and each of them may take one of M different values. Then the random vector $(F_1, \ldots, F_k, C)$ has M^k possible states.

It can be shown that

$$\begin{aligned} & I(F_1, \ldots, F_k; C) - I(F_1, \ldots, F_{k-1}; C) \\ = \; & I(F_k; C | F_1, \ldots, F_{k-1}) \qquad (1) \end{aligned}$$

in which $I(F_k; C|F_1, \ldots, F_{k-1})$ is conditional MI (CMI), which quantifies the shared information between F_k and C, given features $F_1, \dots, F_{k-1}$. Equation (1) shows that CMI measures the information gained by appending a new feature F_k to the original $k-1$ features. Because

$$I(F_k; C|F_1, \dots, F_{k-1}) >= 0$$

so

$$I(F_1, \dots, F_k; C) >= I(F_1, \dots, F_{k-1}; C)$$

which suggests adding a feature F_k will never decrease the mutual information. When F_k and C are conditionally independent given features $F_1, \dots, F_{k-1}$, the CMI, $I(F_k; C|F_1, \dots, F_{k-1})$, is zero. In this case, F_k provides no information to predict a category when other features $F_1, \dots, F_{k-1}$ are known. We can see that CMI provides a useful way to select a new feature F_k, since both its relationship with other features and its individual power are expressed. Therefore, another approach could be used to estimate JMI by applying the equation (1).

$$\begin{aligned} & I(F_1, \dots, F_k; C) \\ = \; & I(F_1, \dots, F_{k-1}; C) + I(F_k; C|F_1, \dots, F_{k-1}) \end{aligned}$$

Assume that currently k-1 selected features maximize the JMI $I(F_1, ..F_{k-1}; C)$. Then the next feature, which maximizes the CMI $(F_k; C|F_1, \dots, F_{k-1})$, should be chosen into the feature set to ensure the maximization of the JMI of k features. Thus, features can be selected one by one into the feature set through an iterative process. In each step, a feature F^* is selected only if $I(F^*; C|F_1, \dots, F_{k-1})$ is highest given $k-1$ already picked features. In the beginning, a feature which maximizes the MI $I(F_i; C)$ is first selected into the set. Therefore, choosing a set of features simultaneously by maximizing the JMI is converted to picking features one by one by maximizing the CMI. This approach indicates that F^* is good only if it carries information about C, and this information has not been captured by any of the features already picked. Thus, elements in a feature set tend to be independent of each others. But unfortunately, maximization of the CMI encounters the same problem as the JMI. It also suffers from the computationally intractable problem when the feature number increases. In the following, we propose an algorithm to overcome this difficulty.

2.2 CMIM Algorithm

To directly calculate CMI, $I(F^*; C|F_1, \dots, F_k)$, we need to compute the complex joint probability, which would be both computationally intractable, and not robust. To avoid this difficulty, we would like to evaluate CMI by decoupling it into more simple forms without the involvement of complex joint probability. We first try to approximate $I(F^*; C|F_1, \dots, F_k)$ with k fewer dimensional forms such as $I(F^*; C|\underbrace{F_i, \dots, F_j}_{k-1})$. Because more information will degrade uncertainty, $I(F^*; C|F_1, \dots, F_k)$ is certain to be smaller than any CMI with fewer dimensional forms $I(F^*; C|\underbrace{F_i, \dots, F_j}_{k-1})$. Therefore, we estimate $I(F^*; C|F_1, \dots, F_k)$ by their minimum value, i.e.,

$$I(F^*; C|F_1, \dots, F_k) \approx \min I(F^*; C|\underbrace{F_i, \dots, F_j}_{k-1}) \qquad (2)$$

Table 1: Conditional Mutual Information Maximin (CMIM) Algorithm

```
Input: $n$ - the number of features to be selected
         $v$ - the number of total features
Output: $\boldsymbol{F}$ – the set for selected features
1. set $\boldsymbol{F}$ to be empty
2. $m = 1$ ($m$ - counter)
3. add $F_i$ in $\boldsymbol{F}$, where $F_i = \arg\max_{i=1..v} I(F_i; C)$
4. repeat
5.     $m++$
6.     Add $F_i$ in $\boldsymbol{F}$, where
           $F_i = \arg\max_{i=1..v}\{\min_{F_j \in \boldsymbol{F}} I(F_i; C|F_j)\}$
7. until $m = n$
```

The $k-1$ features minimizing the CMI in (2) are the ones most correlated with the feature F^* in the selected features. So the predictive power of F^* is undermined. To avoid this circumstance, we need to select a feature to make $\min I(F^*; C|\underbrace{F_i, \dots, F_j}_{k-1})$ as large as possible. When a feature F^* is weakly influenced by the features already picked and itself is important, such a feature should maximizes $\min I(F^*; C|\underbrace{F_i, \dots, F_j}_{k-1})$. Thus, we ensure that a new feature is both informative and orthogonal to the preceding ones.

In this paper, we use the triplet form $I(F^*; C|F_i)$ to estimate the CMI $I(F^*; C|F_1, \dots, F_k)$, which will greatly relieve the computation overhead. This simpler form quantifies information that the feature F^* encodes to predict the category, given a single feature F_i. We replace the right form in equation (2) with the more simpler triplet $I(F^*; C|F_i)$, i.e.

$$I(F^*; C|F_1, \dots, F_k) \approx \min_i I(F^*; C|F_i) \qquad (3)$$

We now need to select a feature F^* maximizing $min(I(F*; C|F_i))$. Hence, we get our algorithm called conditional mutual information maximin (CMIM). This algorithm is shown in Table 1.

In the algorithm, we approximate the true CMI by triplet form $I(F^*; C|F_i)$. We can, of course, use a more complex form, such as quadruplet $I(F^*; C|F_i, F_j)$, to estimate the true CMI in a similar way. However there are totally $k(k-1)/2$ quadruplets for k selected features and the computation for each quadruplet is more complex than for a triplet. Consequently approximations based on more complex forms always cause more severe efficiency problems. Although complex forms seem to be able to acquire a more accurate estimate, due to the sparse data problem, they always need to use smoothing techniques to estimate the joint probability. Therefore, our algorithm, based on the triplet, can not only effectively and robustly avoid features' dependence on each other, but also it is more attractive in terms of efficiency.

The computational cost of CMIM is much smaller than the cost for JMI. We denote here V as the total number of terms in documents, N as the number of features we want to select, and D as the size of documents for training. For a certain F^*, we need to compute $I(C; F^*|F_j)$ for all F_j in $\boldsymbol{F}$, which costs $|\boldsymbol{F}|$ iterations. Selecting each feature will cost $\sum_{i=1,\dots,V}(V - |\boldsymbol{F}|)|\boldsymbol{F}| = O(V^3)$iteration. Therefore, the total cost of CMIM is $O(NV^3)$. This is true as long as we build a table to store the triplet joint probability $p(c, f_i, f_j)$ for all categories and features, which requires $O(D)$ operations and $O(V^2)$ space complexity as a preprocessing step.

2.3 Related Work

Information theory has been widely investigated for a long time, and many methods based on it have been used to analyze data. Entropy and mutual information are the most basic concepts in information theory, and many researchers have investigated them from many aspects. However the concept – "conditional" – for mutual information is relatively new. Two other studies have been performed using conditional mutual information. The first, a study by [22] used the conditional mutual information as feature selection methods, but it focused on how to estimate complex joint probabilities to compute the CMI. [8] proposed the conditional mutual information maximization for feature selection, and compared it with mutual information, which is used for face recognition. As shown in [24], mutual information is not a good choice for feature selection in text categorization. Therefore we will compare CMIM with information gain in our experiment.

3. INFORMATION GAIN AND CLASSIFIERS

3.1 Information Gain

Information gain is a popular feature selection method, frequently employed as a term goodness criterion in the field of machine learning. It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document. Let $\{c_i\}_{i=1}^m$ denote the set of categories in the target space. The information gain of a term t is defined to be

$$\begin{aligned} G(t) = & -\sum_{i=1}^{m} Pr(c_i) \log Pr(c_i) \\ & + Pr(t) \sum_{i=1}^{m} Pr(c_i|t) \log Pr(c_i|t) \\ & + Pr(\bar{t}) \sum_{i=1}^{m} Pr(c_i|\bar{t}) \log Pr(c_i|\bar{t}) \end{aligned}$$

Given a training corpus, we compute the information gain for each unique term to obtain a ranked term list. For each term, it costs a constant operations given a certain number of categories. The total overhead has a time complexity of $O(V)$, while the probability estimation has a time complexity of $O(D)$ and a space complexity of $O(V)$, where V is the vocabulary size, and D is the size of the training cases.

3.2 Naïve Bayes

Although simple, naïve Bayes (NB) models have been remarkably successful in information retrieval. It has been shown that even if the independence assumption is violated, the NB classifier will be optimal [6]. We use an NB classifier in our experiment. The NB classifier estimates the probability of each category given features of a new document. The Bayesian equation is to estimate the probabilities:

$$P(C = c_k|\vec{f}) = \frac{P(\vec{f}|C = c_k)P(C = c_k)}{P(\vec{f})} \quad (4)$$

The quantity $P(\vec{x}|C = c_k)$ is often impractical to compute, so we assume the features are conditionally independent, given the category variable C. This simplifies the computation to

$$P(\vec{f}|C = c_k) = \prod_i P(f_i|C = c_k) \quad (5)$$

where f_i represents the terms in the documents.

3.3 Support Vector Machines

Vapnik [20] proposed the theory and algorithm of support vector machines. Since then, this method has been gaining popularity in the machine learning community. It was first applied to text categorization by Joachims [10]. SVM is a method for efficiently training a linear classifier in the feature space. This method first maps the documents into a high dimensional feature space, and then tries to learn a separating hyperplane for maximizing the margins between different kinds of documents. Of course, not all problems are linearly separable. This difficulty can be solved by a modification of the optimization formulation that allows, but penalizes, examples that fall on the wrong side of a decision boundary.

Training a SVM requires the solution of a quadratic programming (QP) problem. Any QP optimization method can be used to learn the weights on the basis of training examples. However, many QP methods can be very slow for a large-scale data set. Fortunately, Platt proposed a sequential minimal optimization (SMO) algorithm [16] for training a support vector classifier quickly. Although SVMs can only solve the binary classification problem, multi-class problems can be solved using pairwise classification with good performance.

4. EXPERIMENTAL RESULT

In the experimental evaluation, we focus on a comparison between two feature selection methods: the conditional mutual information maximin (CMIM) and information gain (IG). We use two classifiers: naïve Bayes and SVMs. The results are based on two different data sets: WebKB and Newsgroups.

4.1 WebKB: Web Pages Collection

The WebKB data set [4] contains web pages gathered from several universities' computer science departments. The pages are divided into seven categories: student, faculty, staff, course, project, department and other. In this paper, we use the four most populous entity-representing categories: student, faculty, course, and project, which all together contain 4199 pages.

A held-out set with 20% of the data was created randomly. The other 80% were used as training data. Before feature selection was applied on the training set, we first performed stop word removal, stemming, and case-conversion to lower case. Reserving those terms occurring at least six times in a corpus, we have 3359 training documents with a vocabulary size of 7161. Note that the document number among categories is not uniformly distributed: i.e. documents in project (504 pages) are fewer than 1/8 in proportion of the total documents.

Figures 1 and 2 show the result of the classification accuracy using *micro averaging.* Here we use a logarithm scaled x-axis to emphasize the result when the feature count is

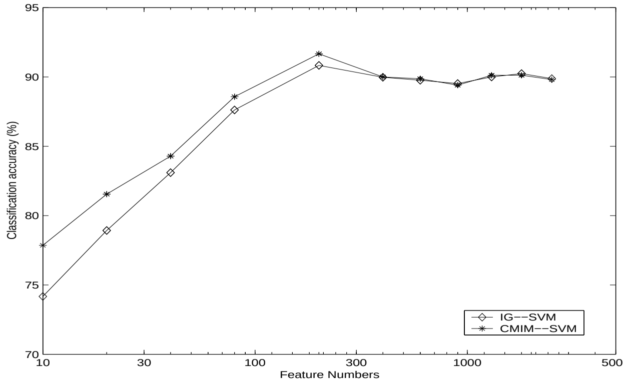

Figure 1: Micro-averaged classification accuracy vs. Feature size, using SVM as classifier

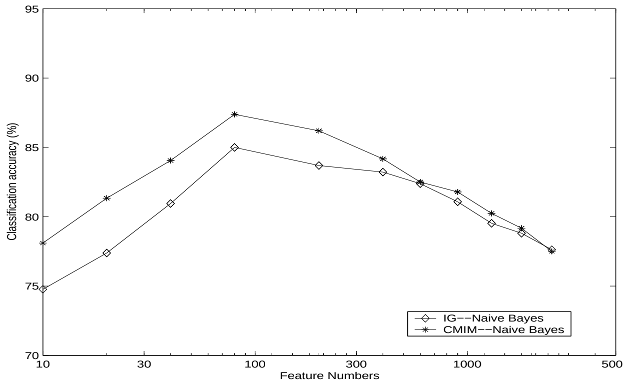

Figure 2: Micro-averaged classification accuracy vs. Feature size, using naïve Bayes as classifier

small. The performance of SVM and naïve Bayes are generally similar, except when the feature size is large, naïve Bayes performs worse than SVM. When the feature count is small (less than 100), CMIM clearly shows better performance than IG. As the feature count gets larger, the difference in accuracy vanishes. From the evaluation result we conclude that CMIM can provide a modest increase ($\leq 5\%$) in micro-averaged accuracy, as compared with IG.

Classification accuracy only gives a rough assessment of how good a classifier is. In order to get a better interpretation of the result, it is useful to know which classes of data were most often misclassified. The confusion matrix contains information about cases for their actual and predicted classes. In this matrix, all the columns represent the predicted classes, and thus a piece of data belongs to the column if it is classified as belonging to this class. The rows represent the actual classes, and a piece of data is in a particular row if it belongs to the corresponding class. A perfect classification outcome results in a matrix with 0's everywhere except on the diagonal. Table 2 selects some confusion matrices with a series of feature numbers.

The first row illustrates classification using information gain. In this row, we can see, in the first cell, the row for category **c** is (2, 15, 0, 90) when the feature size is 10. The element in the third position is zero, which means all cases in category **c** are predicted to the wrong categories; i.e. 2 cases for **a**, 15 cases for **b**, and 90 case for **d**. Therefore, the per-category classification is zero for category **c**. This demonstrates that when the feature size is small, representative features of category **c** have not yet been detected by the information gain criterion. Most cases in other categories have been assigned to their correct classes, so their diagonal elements are much larger than the elements in other cells. In contrast, the second row in this table, which is based on CMIM , is greatly different. In the first matrix, the diagonal element for category **c** is 22 , which is much greater than zero as compared with information gain.

When the feature size becomes 20, category **c** has been detected using IG. But it gets only 22.4% per-category accuracy for category **c**, but 54.4% for CMIM already. Therefore, representative features in the minor category **c** have been much better detected by CMIM. Table 2 shows the perceptible differences between CMIM and IG. Classification based on IG is inclined to classify the cases in major categories, while CMIM tends to evenly deal with each category.

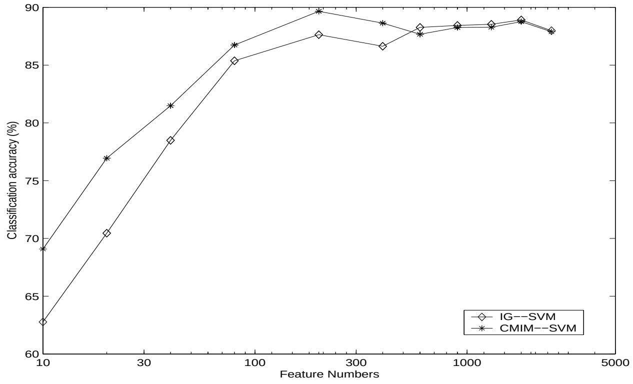

Figure 3: Macro-averaged classification accuracy vs. Feature size, using SVM as a classifier

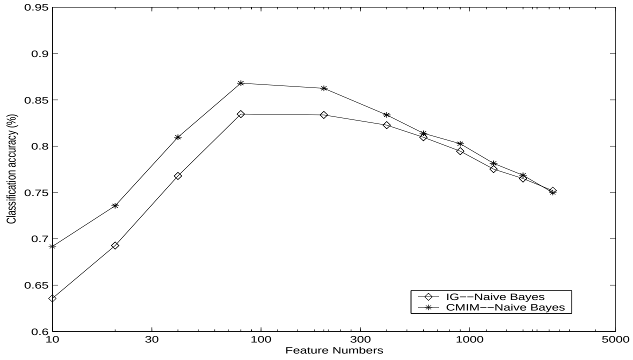

Figure 4: Macro-averaged classification accuracy vs. Feature size, using naïve Bayes as classifer

Micro-averaged accuracy gives equal weight to every document, and therefore tends to over-emphasize the performance on the largest categories. The *macro-average scores*, on the contrary, give equal weight to every category. Macro-averaged accuracy scores are computed by first computing

Table 2: Confusion matrices based on IG and CMIM, using SVM as a classifier. (a - course, b - faculty, c - project, d - student)

	10 features					20 features					40 features				
IG	a	b	c	d		a	b	c	d		a	b	c	d	
	149	10	0	22	— a	155	7	1	18	— a	165	3	1	12	— a
	4	167	0	44	— b	3	179	6	27	— b	2	183	6	24	— b
	2	15	**0**	90	— c	2	14	**24**	67	— c	4	13	**53**	37	— c
	8	22	0	307	— d	5	24	3	305	— d	5	25	10	297	— d
CMIM	a	b	c	d		a	b	c	d		a	b	c	d	
	151	8	0	22	— a	155	7	1	18	— a	158	5	3	15	— a
	4	176	7	28	— b	4	177	7	27	— b	7	179	8	21	— b
	1	15	**22**	69	— c	2	10	**55**	40	— c	2	5	**71**	29	— c
	6	23	3	305	— d	7	23	9	298	— d	7	18	12	300	— d

the accuracy scores for each per-category row in the confusion matrix, and then averaging these per-category scores to compute the global means. We show the result for macro-averaged accuracy in Figure 3 and Figure 4. We can derive the result from the figures that CMIM gives a remarkable increase ($\leq 10\%$) against IG.

4.2 Newsgroups: Discussion Articles Collection

The Newsgroups data set contains about 20000 articles evenly divided among 20 UseNet discussion groups [11]. Many of the categories fall into confusable clusters; for example, five of them are comp.* discussion groups, and three of them are religion. When tokenizing this data, we skip the UseNet headers, and subject line. In section 2, we gave an intuitive argument that CMIM will perform much better than IG when the category number is not uniformly distributed and the feature size is small. To evaluate this idea, we manually constructed the data set as follows: Building an arithmetic sequence from 50 to 1000 with difference 50, we randomly assigned each number of this sequence to a category as the number of documents selected from this category. Hence in our experiment, the number of the largest category is 1000, which is 20 times larger than 50, the number of the smallest category. We also use 20% held-out as testing data, and the other 80% as training data in this experiment.

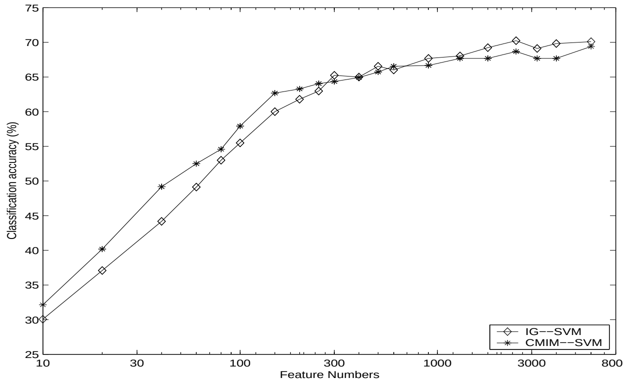

Figure 5: Micro-averaged classification accuracy vs. Feature size, using SVM as classifer

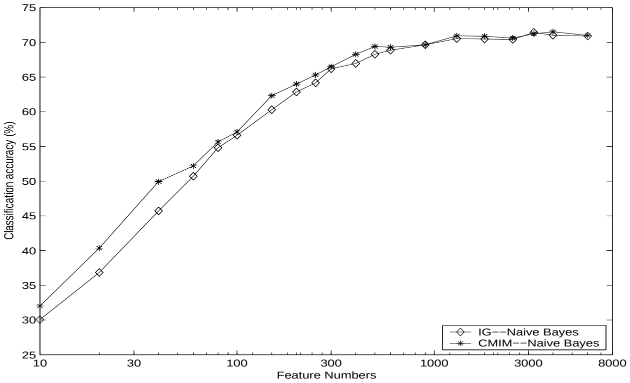

Figure 6: Micro-averaged classification accuracy vs. Feature size, using naïve Bayes as classifer

Figure 5 and Figure 6 show the micro-averaged accuracy between CMIM and IG under SVM, and under naïve bayes. In Figure 5, the classification accuracy using CMIM is slightly lower than MI as feature size is larger than about 500 features. Nevertheless, in overall, CMIM outperforms IG, and provides a moderate ($\leq 5\%$) increase when the feature size is comparatively small.

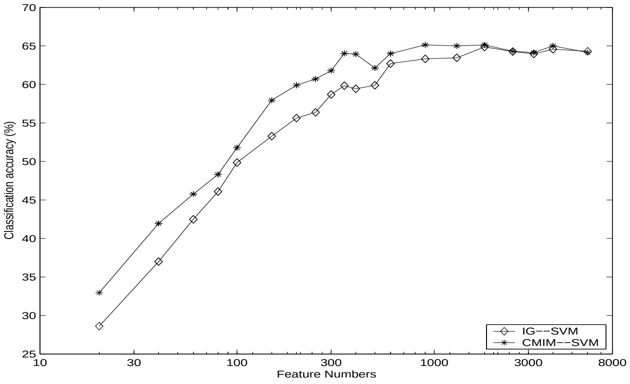

Figure 7: Macro-averaged classification accuracy vs. Feature size, using SVM as classifer

CMIM may reduce the micro-averaged accuracy when the deviation of category number is large. CMIM is liable to select features evenly distributed in each category. Since some categories only take a small portion of total cases, balancing selected features for minor categories sometimes may lower the micro-averaged accuracy. But this will not hap-

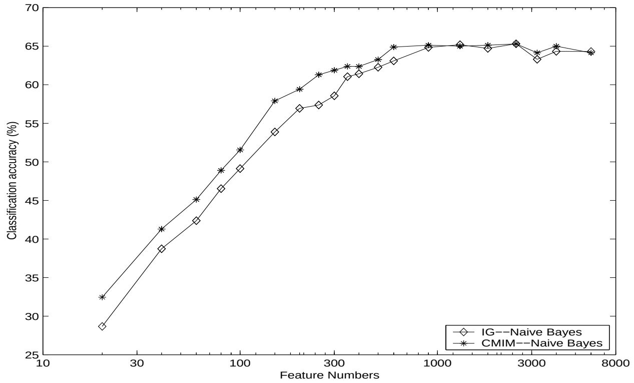

Figure 8: Macro-averaged classification accuracy vs. Feature size, using naïve Bayes as classifer

pen to the macro-averaged classification accuracy, which is shown in Figure 7 and Figure 8. The macro-averaged accuracy based on CMIM gives a remarkable ($\leq 12\%$)accuracy increase.

4.3 Discussion

What is the reason for the different performance between these two feature selection methods, CMIM and IG? We discuss it from three angles: feature size, category number and category deviation.

- IG is based on a feature's individual predictive power, regardless of features' mutual relationship. Selected features have the highest scores individually, which are always representative for a part of the categories, but not for all. So when the feature size is small, the selected features seldom cover all categories. But since representative features for one certain category are always correlated with each other, CMIM can detect such correlation and, thus, bias its focus evenly for all categories. Consequently informative features for minor category are more easily selected by CMIM than by IG. As the selected feature size increases and becomes much larger than the category size, IG will also pick features representative for the minor categories. Thus the difference is not as evident as it is for small feature size.

- The category number is sensitive for IG. With only two or three categories, IG is likely to find features unbiased towards all categories. As the category size increases, the discriminating features for one or two categories are most probably neglected by IG. CMIM, on the contrary, is somewhat immune to category number. It always selects features evenly distributed in different categories.

- Category deviation measures the spread of categories. When the data assigned to different categories deviates greatly, categories can be classified into two classes: the *majority* category and the *minority* category. All feature selection methods without taking consideration of terms relationships, such as information gain and mutual information, are greatly influenced by category deviation, because the predictive power of representative features for the minor categories are nearly completely suppressed by a large quantity of features in the major categories, even if they are only the ordinary features. Therefore some discriminant features are ranked far from their primary position. CMIM is a solution to save such concealed features. It selects an uncorrelated feature with the preceding ones in each step, and tends to focus on the features from major categories to minor ones as feature size increases. Although CMIM is certain to increase the macro-averaged accuracy, it may sometime reduce the micro-averaged accuracy.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we present a novel feature selection method for text categorization. It uses a conditional mutual information maximin algorithm to select features, which are both individually discriminate as well as being dependent on features already selected. The experiments show that both micro-averaged and macro-averaged classification perform better based on this feature selection method, especially when the feature size is small and the category number is large. CMIM is more robust as it requires only estimations of distributions of triplets of features. It provides a good tradeoff between redundancy and individual power of selected features.

CMIM also suffers from some drawbacks. It currently cannot deal with integer-valued or continuous features. It is based on the assumption that dependencies between features can be captured by only a pair of variables. CMIM, therefore, ignores the dependencies among three or larger families of features. Although CMIM has greatly relieved the computation overhead, the complexity $O(NV^3)$ is still not very attractive. Therefore our future work will address these issues. We will consider parameter density models to deal with continuous features, and investigate other conditional models to efficiently formulate features' mutual relationship.

6. ACKNOWLEDGEMENT

The authors would like to thank the anonymous referees for their comments which helped improve the paper.

7. REFERENCES

[1] M. Berry, S. Dumais, and G. W. O. Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 1995.

[2] M. Caropreso, S. Matwin, and F. Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In *Text Databases and Document Management: Theory and Practice*. Idea Group Publishing, Hershey.

[3] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.

[4] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98)*, 1998.

[5] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic indexing. *Journal of the American Socity for Information Science*, 1990.

[6] P. Domingos and M. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 1997.

[7] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19:1, 1993.

[8] F. Fleuret. Binary feature selection with conditional mutual infomration. *Technical Report*, 2003.

[9] L. Galavotti, F. Sebastiani, and M. Simi. Experiments on the use of feature selection and negative evidence in automated text categorization. In *Procedings 4th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'00)*, 2000.

[10] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Procedings 10th European Conference on Machine Learning (ECML'98)*, 1998.

[11] T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *Procedings 4th European Conference on Reaserch and Advanced Technology for Digital Libraries (ECDL'00)*, 2000.

[12] D. Lewis and M. Ringuette. Comparison of two learning algorithms for text categorization. In *Procedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94)*, 1994.

[13] D. Mladenic. Feature subset selection in text learning. In *Procedings 10th European Conference on Machine Learning (ECML'98)*, 1998.

[14] I. Moulinier. Is learning bias an issue on the text categorization problem? *Technical report*, 1997.

[15] K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*, 1999.

[16] J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods*, 2000.

[17] H. Schutze, D. Hull, and J. Pedersen. A comparison of classifiers and document representations for the routing problem. In *Procedings 18th ACM International Conference on Research and Development in Information Retrieval (SIGIR'95)*, 1995.

[18] F. Sebastiani. Machine learning in automated text categorisation. *Technical Report B4-31*, 1999.

[19] I. Tsamardinos, C. Aliferis, and A. Statnikov. Algorithms for large scale markov blanket discovery. In *Proceedings of the 16th International Florida Artificial Intelligence Research Society Conference(FLAIRS)*, 2003.

[20] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, 1995.

[21] E. Wiener, J. Pedersen, and A. Weigend. A neural network approach to topic spotting. In *Procedings of the 4th Annual Symposium on Document Analysis and Information Retrieval (SDAIR'95)*, 1995.

[22] H. Yang and J. Moody. Feature selection based on joint mutual information. In *Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis*, 1999.

[23] Y. Yang and C.G.Chute. A linear least square fit mapping method for information retrieval from natural language texts. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING'92)*, 1992.

[24] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Learning (ICML'97)*, 1997.