Information Processing and Management 48 (2012) 741–754

Contents lists available at SciVerse ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

A new feature selection based on comprehensive measurement both in inter-category and intra-category for text categorization

Jieming Yang [a,b], Yuanning Liu [a], Xiaodong Zhu [a,*], Zhen Liu [a,c], Xiaoxu Zhang [a]

[a] *College of Computer Science and Technology, Jilin University, Changchun, Jilin, China*
[b] *School of Information Engineering, Northeast Dianli University, Jilin, Jilin, China*
[c] *Graduate School of Engineering, Nagasaki Institute of Applied Science, Nagasaki-shi, Nagasaki, Japan*

ARTICLE INFO

Article history:
Received 9 May 2011
Received in revised form 28 November 2011
Accepted 21 December 2011
Available online 18 January 2012

Keywords:
Feature selection
Text categorization
Support Vector Machines
Naïve Bayes

ABSTRACT

The feature selection, which can reduce the dimensionality of vector space without sacrificing the performance of the classifier, is widely used in text categorization. In this paper, we proposed a new feature selection algorithm, named CMFS, which comprehensively measures the significance of a term both in inter-category and intra-category. We evaluated CMFS on three benchmark document collections, 20-Newsgroups, Reuters-21578 and WebKB, using two classification algorithms, Naïve Bayes (NB) and Support Vector Machines (SVMs). The experimental results, comparing CMFS with six well-known feature selection algorithms, show that the proposed method CMFS is significantly superior to Information Gain (IG), Chi statistic (CHI), Document Frequency (DF), Orthogonal Centroid Feature Selection (OCFS) and DIA association factor (DIA) when Naïve Bayes classifier is used and significantly outperforms IG, DF, OCFS and DIA when Support Vector Machines are used.

1. Introduction

As the number of digital documents available on the Internet has been growing significantly in recent years, it is impossible to manipulate manually such enormous quantities of information. More and more methods based on statistical theory and machine learning have been applied to information automatic processing (Shang, Huang, & Zhu, 2007). A very efficient method for managing the vast amount of data is text categorization which assigns one or more predefined categories to one new document based on the contents of the document (Fragoudis, Meretakis, & Likothanassis, 2005; Sebastiani, 2002; Yang & Pedersen, 1997). There exist numerous sophisticated algorithms that have been applied to text categorization; for example, Naïve Bayes classifier (NB) (Chen, Huang, Tian, & Qu, 2009), Support Vector Machines (SVMs) (Joachims, 1998), K-Nearest Neighbors (KNN) (Cover & Hart, 1967), Decision tree, Rocchio (Sebastiani, 2002), etc.

The major characteristic of text categorization is that the number of the features in the feature space (vector space, bag of words) can easily reach the orders of tens of thousands even for moderate size data set (Fragoudis et al., 2005; Yang & Pedersen, 1997). So there exist two problems in the context of the high dimensionality. The one is that some sophisticated algorithms can not be optimally used in the text categorization. The other is that the overfitting is inevitable in the text categorization when most algorithms are trained in the training set (Fragoudis et al., 2005; Sebastiani, 2002). Therefore, dimensionality reduction has been a major research area.

[*] Corresponding author. Tel.: +86 431 85159376.
E-mail address: zhuxiaodong9@gmail.com (X. Zhu).

0306-4573/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2011.12.005

The goal of the dimensionality reduction is to reduce vector space and avoid the overfitting without sacrificing the performance of the categorization, and it is tackled by feature extraction and feature selection. The feature extraction generates a new term set that is not of the same type of the original feature space by combinations or transformations of the original one; however, the feature selection, which is the most commonly used method in the field of text classification, selects a subset from the original feature space according to one evaluation criteria (Sebastiani, 2002). There are three distinct ways of viewing feature selection (Blum & Langley, 1997). the first one is embedded approach that the feature selection process is clearly embedded in the basic induction algorithm; the second one is wrapper approach that selects term subset using the evaluation function that exists as a wrapper around the classifier algorithm, and these features will be used on the same classifier algorithm (John, Kohavi, & Pfleger, 1994; Mladenic & Grobelnik, 2003); the last one is the filter approach that selects the feature subset from the original feature space using one evaluation function which is independent to the classifier algorithm (Mladenic & Grobelnik, 2003). As the filter feature selection approach is simple and efficient, it has been widely applied in the text categorization. The proposed method in this study is also a filter approach. There are numerous efficient and effective feature selection algorithms, such as Document Frequency (DF) (Yang & Pedersen, 1997), DIA association factor (DIA) (Fragoudis et al., 2005; Fuhr et al., 1991; Sebastiani, 2002), Odds Ratio (OR) (Mengle & Goharian, 2009), Mutual Information (MI) (Peng, Long, & Ding, 2005; Yang & Pedersen, 1997), Information Gain (IG) (Ogura, Amano, & Kondo, 2009; Yang & Pedersen, 1997), Chi-square statistics (CHI) (Ogura et al., 2009; Yang & Pedersen, 1997), Ambiguity Measure feature selection (AM) (Mengle & Goharian, 2009), Orthogonal Centroid Feature Selection (OCFS) (Yan et al., 2005), Improved Gini index (GINI) (Mengle & Goharian, 2009), Expected Crossed Entropy (Koller & Sahami, 1997), Best Terms (BT) (Fragoudis et al., 2005), measure using Poisson distribution (Ogura et al., 2009), Preprocess algorithm for filtering irrelevant information Based on the Minimum Calss Difference (PBMCD) (Chen & Lü, 2006), Class Dependent Feature Weighting (CDFW) (Youn & Jeong, 2009), binomial hypothesis testing (Bi-Test) (Yang, Liu, Liu, Zhu, & Zhang, 2011), and so on.

Among the above-mentioned feature selection algorithms, Document Frequency (DF) only measures the significance of a term in the intra-category; however, Ambiguity Measure (AM) and DIA association factor (DIA) only calculate the score of a term in the inter-category. In this paper, we proposed a new feature selection algorithm, called Comprehensively Measure Feature Selection (CMFS), which comprehensively measures the significance of a term both in the intra-category and inter-category. To evaluate the CMFS method, we used two classification algorithms, Naïve Bayes (NB) and Support Vector Machines (SVMs) on three benchmark corpora (20-Newgroups, Reuters-21578 and WebKB), and compared it with six feature selection algorithms (Information Gain, Chi-square statistic, Improved Gini index, Document Frequency and Orthogonal Centroid Feature Selection, DIA association factor). The experiment results show that the proposed method CMFS outperforms significantly DIA, IG, CHI, DF, OCFS, and is comparable with GINI when Naïve Bayes is used; the CMFS is superior to DIA, IG, DF and OCFS, and is comparable with GINI and CHI when Support Vector Machines is used.

The rest of this paper is organized as follows: Section 2 presents the state of the art for the feature selection algorithms. Section 3 describes the basic idea and implementation of the CMFS method. The experimental details are given in Section 4 and the experimental results are presented in Section 5. Section 6 shows the statistical analysis and discussion. Our conclusion and the future work direction are provided in the last section.

2. Related work

Numerous feature selection methods have been widely used in text categorization in recent years. The Information Gain (IG) and Chi-square statistic (CHI) are two of the most efficient feature selections, and Document Frequency (DF) is comparable to the performance of IG and CHI (Y. Yang & Pedersen, 1997). Improved Gini index is an improved feature selection based on Gini index. It has a better performance and simpler computation than IG and CHI (Shang et al., 2007). The Orthogonal Centroid Feature Selection (OCFS) was proposed by Yan et al. (2005). It was designed by optimizing the objective function of the effective Orthogonal Centroid algorithm. The experimental results show that OCFS is more efficient than the popular IG and CHI methods. The underlying premise under the Ambiguity Measure (AM) is the quick identification of unambiguous terms (Mengle & Goharian, 2009). Mengle and Goharian (2009) thought the features that point to only one category perform better than features that point to more than one category. Its results indicated that AM feature selection improved over Odds Ratio (OR), Information Gain (IG), Chi-square statistic (CHI). DIA association factor (DIA) was proposed in AIR/X system by Fuhr et al. (1991). The idea of DIA is similar to the AM. In this section, the six feature selections used in this paper are detailed as follows.

2.1. Improved Gini index

Gini index is a non-purity split method and widely used in decision tree algorithms. To apply the Gini index directly to the text feature selection, Shang et al. (2007) proposed the improved Gini index method. It measures the purity of feature t_k toward a category c_i. The bigger the value of purity is, the better the feature is. The formula of the improved Gini index is defined as follows:

$$\mathrm{Gini}(t_k) = \sum_i P(t_k|c_i)^2 P(c_i|t_k)^2$$

where $P(t_k|c_i)$ is the probability that the feature t_k occurs in category c_i; $P(c_i|t_k)$ refers to the conditional probability that the feature t_k belongs to category c_i when the feature t_k occurs.

2.2. Information Gain

Information Gain (IG) (Quinlan, 1986) is frequently used as a criterion in the field of machine learning (Yang & Pedersen, 1997). The Information Gain of a given feature t_k with respect to the class c_i is the reduction in uncertainty about the value of c_i when we know the value of t_k. The larger Information Gain of a feature is, the more important the feature is for categorization. Information Gain of a feature t_k toward a category c_i can be defined as follows:

$$\mathrm{IG}(t_k, c_i) = \sum_{c\in\{c_i,\bar{c}_i\}} \sum_{t\in\{t_k,\bar{t}_k\}} P(t,c)\log\frac{P(t,c)}{P(t)P(c)}$$

where $P(c)$ is the fraction of the documents in category c over the total number of documents, $P(t,c)$ is the fraction of documents in the category c that contain the word t over the total number of documents. $P(t)$ is the fraction of the documents containing the term t over the total number of documents (Youn & Jeong, 2009).

2.3. Chi-square

χ^2-statistic (Chi-square) testing (Yang & Pedersen, 1997) was used to test the independence of two variables in mathematical statistics. We used Chi-square testing to determine the independence of the feature t_k and the category c_i. If $\chi^2(t_k,c_i) = 0$, the feature t_k and the class c_i are independent, so the feature t_k does not contain any category information. Otherwise, the greater the value of the $\chi^2(t_k,c_i)$ is, the more category information the feature t_k owns. Chi-square formula is defined as follows:

$$\chi^2(t_k, c_i) = \frac{N(a_{ki}d_{ki} - b_{ki}c_{ki})^2}{(a_{ki}+b_{ki})(a_{ki}+c_{ki})(b_{ki}+d_{ki})(c_{ki}+d_{ki})}$$

where N is the total number of messages; a_{ki} is the frequency that feature t_k and category c_i co-occur; b_{ki} is the frequency that feature t_k occurs and does not belong to category c_i; c_{ki} is the frequency that category c_i occurs and does not contain feature t_k; d_{ki} is the number of times neither c_i nor t_k occurs.

2.4. Document Frequency

Document Frequency (DF) is a simple and effective feature selection method, and it computes the number of documents in which a term occurs. The basic idea is that the rare terms are not useful for category prediction and maybe degrade the global performance (Yang & Pedersen, 1997). So if the number of the documents in which a term occurs is the largest, the term is retained (Sebastiani, 2002). The document frequency of a term is calculated as follows:

$$\mathrm{DF}(t_k, c_i) = P(t_k|c_i)$$

2.5. Orthogonal Centroid Feature Selection

The Orthogonal Centroid Feature Selection (OCFS) selects features optimally according to the objective function implied by the Orthogonal Centroid algorithm (Mengle & Goharian, 2009; Yan et al., 2005). The centroid of each class and all training samples are firstly calculated, and then the score of the term is calculated according to the centroid of the each class and the entire training set. The higher the score of the term is, the more category information the term contains. The score of a term t_k is calculated as follows:

$$\mathrm{OCFS}(t_k) = \sum_{j=0}^{|C|} \frac{n_j}{n}(m_j^k - m^k)^2$$

where n_j is the number of documents in the category c_j, n is the total number of documents in the training set, m_j^k is the kth element of the centroid vector m_j of category c_j, m^k is the kth element of the centroid vector m of entire training set, $|C|$ refers to the total number of categories in the corpus.

2.6. DIA association factor

DIA association factor (Fuhr et al., 1991; Sebastiani, 2002) is an important tool in automatic indexing. It is an estimate of the probability for the category c_i to be assigned to a document if this document contains the term t_k. The DIA association factor determines the significance of the occurrence of the term t_k with respect to the category c_i. The DIA association factor is defined by

$$\mathrm{DIA}(t_k, c_i) = P(c_i|t_k)$$

where $P(c_i|t_k)$ refers to the conditional probability that the feature t_k belongs to category c_i when the feature t_k occurs.

3. Algorithm description

3.1. Motivation

Before the process of text classification, the feature vector space model (bag of words) (Sebastiani, 2002), which consists of all unique terms extracted from the training set, must be created. Then, each raw document in the corpus must be transformed into a big vector according to mapping of the terms occur in the raw document to the feature vector space. This vector space model was regarded as a word-to-document feature-appearance matrix where rows are the features and columns are document vectors (Chen & Lü, 2006). It describes the occurrences of one feature in every document in the corpus. Differently to this approach, as explained in this paper, we consider the significance of one feature based on the occurrences in every category in the corpus. The number of categories is predefined in text classification (Sebastiani, 2002), so we regard this vector space model as a term-to-category feature-appearance matrix where rows are the features and columns are category vector. The elements in the term-to-category matrix are the number of documents in which a feature occurs in every category. Table 1 shows a subset of term-to-category feature appearance matrix for top 10 categories of Reuters-21578 corpus. In fact, the above mentioned term-to-category matrix is the basis of the most feature selections. The Document Frequency (DF) (Yang & Pedersen, 1997) only computes the document frequency of each unique term in one category, and then the highest document frequency of a term in various categories is retained as the term's score. The DIA association factor (DIA) (Fragoudis et al., 2005; Sebastiani, 2002) and Ambiguity Measure (AM) (Mengle & Goharian, 2009) only calculate the distribution probability of a term in various categories, and then the highest probability of the term be used as the term's score. Thus the DF method concentrates on the column of the term-to-category matrix, while DIA and AM focus on the row of the term-to-category matrix. An interesting phenomenon was paid attention to during comparing the performance of the feature selections. Both DF and DIA only focus on one respect of the problem (row or column). For example, if we select 5 features from the vector space model listed in Table 1, the features selected by DF and DIA are completely different. Table 2 shows the features selected by DF, DIA, CMFS and their scores calculated with DF, DIA, CMFS, respectively. Moreover, the category which the selected terms stand for are list in Table 2. It can be seen from Table 2 that the features, which contain more information for categorization judged by DF, are irrelevant features in the view of DIA, vice versa. Based on this observation, a new feature selection algorithm, called Comprehensively Measure Feature Selection (CMFS), is proposed in this paper. It comprehensively measures the significance of a term both in inter-category and intra-category.

3.2. Algorithm implementation

In this paper, we propose a hybrid strategy that comprehensively measures the significance of a term. It means that CMFS calculates the significance of a term from both inter-category and intra-category. Thus, we define comprehensive measurement for each term t_k with respect to category c_i as follows:

$$\mathrm{CMFS}(t_k, c_i) = \frac{tf(t_k, c_i) + 1}{tf(t_k) + |C|} \bullet \frac{tf(t_k, c_i) + 1}{tf(t, c_i) + |V|} = \frac{(tf(t_k, c_i) + 1)^2}{(tf(t_k) + |C|)(tf(t, c_i) + |V|)} \tag{1}$$

where $tf(t_k,c_i)$ is the frequency of a term t_k in category c_i; $tf(t_k)$ is the frequency of a term t_k in the entire training set; $tf(t,c_i)$ is the sum of frequency of all terms in category c_i; $|C|$ is the number of the categories; $|V|$ is the total number of terms in the feature vector space. To measure globally the goodness of a term, two alternate ways can be used to combine the category-specific scores of a term (Yang & Pedersen, 1997). The Eq. (3) is used in this paper.

$$\mathrm{CMFS}_{avg}(t_k) = \sum_{i=1}^{|C|} P(c_i) CMFS(t_k, c_i) \tag{2}$$

Table 1
The term-to-category feature appearance matrix.

Features	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Plan	868	36	180	1227	102	66	118	30	155	58
Designer	12	0	0	2	0	0	1	0	0	0
Kits	72	25	47	85	59	20	32	22	62	24
Home	230	9	14	31	20	4	7	4	4	16
Design	216	30	99	65	61	18	30	9	39	33
Projects	321	11	19	28	34	11	35	6	29	27
Sales	80	16	61	175	35	31	38	14	54	15
Terms	697	58	154	415	120	286	239	54	105	50
Disclosed	240	2	2	77	2	6	7	1	0	2
Bank	141	4	10	25	7	3	10	6	8	3

Table 2
The features selected by DF, DIA and CMFS, respectively.

DF			DIA			CMFS		
Features	Score	The category the term stands for	Features	Score	The category the term stands for	Features	Score	The category the term stands for
Terms	0.6307	C6	Disclosed	0.6905	C1	Plan	0.2473	C4
Plan	0.5738	C4	Home	0.6618	C1	Terms	0.0827	C6
Design	0.1677	C3	Bank	0.6255	C1	Projects	0.0676	C1
Kits	0.1474	C8	Projects	0.6064	C1	Disclosed	0.0576	C1
Sales	0.1180	C9	Designer	0.5200	C1	Home	0.0530	C1

$$\mathrm{CMFS}_{\max}(t_k) = \max_{i=1}^{|C|}\{\mathrm{CMFS}(t_k, c_i)\} \tag{3}$$

In the view of the theory of probability, we can regard the part before sign of multiplication in Eq. (1) as the conditional probability that the feature t_k belongs to category c_i when the feature t_k occurs, $P(c_i|t_k)$, and the part after the sign of multiplication in Eq. (1) can be considered as the probability that the feature t_k occurs in category c_i, $P(t_k|c_i)$. So the Eqs. (1)–(3) can be represented as follows:

$$\mathrm{CMFS}(t_k, c_i) = P(t_k|c_i)P(c_i|t_k) \tag{4}$$

$$\mathrm{CMFS}_{avg}(t_k) = \sum_{i=1}^{|C|} P(c_i)\mathrm{CMFS}(t_k, c_i) = \sum_{i=1}^{|C|} P(c_i)P(t_k|c_i)P(c_i|t_k) \tag{5}$$

$$\mathrm{CMFS}_{\max}(t_k) = \max_{i=1}^{|C|}\{\mathrm{CMFS}(t_k, c_i)\} \tag{6}$$

Given the term-to-category feature-appearance matrix V. The CMFS algorithm operates in three steps. The first step of the CMFS algorithm aims to compute the significance of a term t_k in one category c_i against other categories based on the appearances of the term occurring in various categories; the second step of the CMFS algorithm aims to compute the significance of the term t_k occurring in one category c_i against other terms in this category; the significances of the term t_k obtained in the first two steps are combined as the contribution of the term t_k for category c_i in third step. The details of the CMFS algorithm are given as following.

Algorithm 1.

```
Input: $V$ – the term-to-category feature-appearance matrix where rows are the features and columns are categories
  $k$ – the number of the selected features
Output: $V_s$ – the feature subset
Step 1: obtains the number of categories (the number of columns in $V$) – $|C|$
Step 2: obtains the size of the feature vector space (the number of rows in $V$) – $|V|$
Step 3: for each column (each category) $c_i$
Step 4:    obtains the sum of frequency of all features in category $c_i$ – $tf(t,c_i)$
Step 5: end for
Step 6: for each row (each feature) $t_k \in V$
Step 7:   obtains the frequency of the feature $t_k$ in the entire training set – $tf(t_k)$
Step 8:   for each column (each category) $c_i$
Step 9:   obtains the frequency of the feature $t_k$ in category $c_i$ – $tf(t_k,c_i)$
Step 10:   calculates the significance of the feature $t_k$ against the category $c_i$ – $CMFS(t_k,c_i)$
Step 11:   obtains the maximum value of $CMFS(t_k,c_i)$ – $CMFS(t_k)$
Step 12: end for
Step 13: end for
Step 14: ranks all features in $V$ based on $CMFS(t_k)$
Step 15: selects top $k$ features into $V_s$
```

The algorithm is implemented using C++ standard template library (STL). The vocabulary is represented by a container (map<int, string>) which stores the features extracted from training set. A data struct named "_category" is used to represent one category. One of its fields is a container (map<int, int>) which stores the occurrence of each feature in the category. All categories are represented by a container (list<struct _category>). It is the term-to-category feature-appearance matrix mentioned above. The algorithm is run on these containers. The significance of each feature is stored in a container (multimap<double, int>) in which all elements are ordered and the top k feature can be selected.

3.3. Time complexity analysis of CMFS

We assume that the term-to-category feature-appearance matrix V is given. Thus, the time complexity analysis of the CMFS indicates that:

- The calculation of the number of categories costs $O(1)$.
- The estimation of the size of the feature vector space costs $O(1)$.
- Estimating the sum of frequency of all features in every category costs $O(|C|\cdot|V|)$.
- Computing the frequency of every feature in the entire training set costs $O(|C|\cdot|V|)$.
- Computing the significance of every feature t_k costs $O(|C|\cdot|V|)$.
- Ranking all features in V based on CMFS(t_k) costs $O(|V|\cdot\log|V|)$

Combining the above costs, we can easily calculate the cost of the CMFS algorithm as follows:

$$O(1+1+|C|\cdot|V|+|C|\cdot|V|+|C|\cdot|V|+|V|\log|V|)=O(2+3|C|\cdot|V|+|V|\cdot\log|V|)$$

Dropping all constants, the time complexity of the CMFS algorithm is $O(|C|\cdot|V| + |V|\cdot\log|V|)$. From formulas of CMFS, DF, DIA and improved Gini index, we find that the time complexity of CMFS is similar to that of improved Gini index and higher than that of DF and DIA. In addition, previous studies have shown that IG, CHI and OCFS are similar in terms of time complexity (Yan et al., 2005; Yang & Pedersen, 1997); DF and improved Gini index are simpler than IG, CHI and OCFS (Shang et al., 2007; Yang & Pedersen, 1997). It is concluded that the time complexity of CMFS is lower than that of IG, CHI and OCFS, similar to improved Gini index and higher than that of DF and DIA.

4. Experimental setup

4.1. Classifiers

In this section, we briefly describe the Naïve Bayes (NB) and Support Vector Machines (SVMs) used in our study.

4.1.1. Naïve Bayes classifier

The Naïve Bayes (McCallum & Nigam, 1998) is a classifiable algorithm based on the assumption that a term occurring in a document is independent from the occurrence of other terms. There are two commonly used models about Bayesian classifier: one is a multinomial model; the other is the multivariate Bernoulli model. Schneider (2004) indicated that multinomial model can generate higher accuracy than multivariate Bernoulli model. In this study, we use multinomial model.

4.1.2. Support Vector Machines

Support Vector Machine was developed by Drucker et al. (1999) for spam categorization and applied to text categorization by Joachims (1998). Compared to the state-of-art methods, the Support Vector Machines is a higher efficient classifier in text categorization. In this study, we use LIBSVM toolkit (Chang & Lin, 2001), and choose linear kernel support vector machine.

4.2. Datasets

In order to evaluate the performance of the proposed method, three benchmark datasets (Reuters-21578, WebKB and 20-Newsgroups) were used in our experiment. In data preprocessing, all words were converted to lower case, punctuation marks were removed, stop lists were used, and no stemming was used. Document frequency of a term was applied in text representation and the 10-fold validation was adopted in this paper.

4.2.1. 20-Newsgroups

The 20-Newgroups were collected by Ken Lang (1995) and has become one of the standard corpora for text categorization. It contains 19997 newsgroup postings, and all documents were assigned evenly to 20 different UseNet groups.

4.2.2. Reuters-21578

The Reuters-21578 contains 21578 stories from the Reuters newswire. All stories are non-uniformly divided into 135 categories. In this paper, we only considered the top 10 categories.

4.2.3. WebKB

The WebKB is a collection of web pages from four different college web sites. The 8282 web pages are non-uniformly assigned to 7 categories. In this paper, we selected 4 categories, “course”, “faculty”, “project” and “student”, as our corpus.

4.3. Performance measures

We measured the performance of the text categorization in terms of F1 and Accuracy (Sebastiani, 2002). The F1 is determined by the classical informational retrieval parameters, "precision" and "recall". Precision is the ratio of the number of messages which are correctly identified as the positive category to the total number of messages which are identified as the positive category. Recall is the ratio of the number of the messages which are correctly identified as the positive category to the total number of the messages which actually belong to the positive category. The precision and recall are defined for per category. The formulas of the precision and recall for the category c_i are defined as

$$P_i = \frac{TP_i}{TP_i + FP_i},$$

$$R_i = \frac{TP_i}{TP_i + FN_i}.$$

where TP_i is the number of the documents that is correctly classified to category c_i, FP_i is the number of the documents that is misclassified to the category c_i, TN_i is the number of the documents that is correctly classified to other categories excluding the category c_i, FN_i is the number of the documents belonging to category c_i were misclassified to other categories. For evaluating performance average across categories, the micro-averaging was used in our experiment. The micro-precision and micro-recall may be obtained as

$$P_{micro} = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FP_i)},$$

$$R_{micro} = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|} (TP_i + FN_i)}.$$

where $|C|$ is the number of the categories. The denominator of formula for micro-precision is constant and equal to the amount of the documents. So the micro-precision, different from the precision on single category, is insensitive to the number of the false positive documents. The micro F1 and Accuracy are defined in the following way:

$$F1_{micro} = \frac{2P_{micro}R_{micro}}{P_{micro} + R_{micro}},$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

5. Results

5.1. Experimental results on the 20-newsgroups dataset

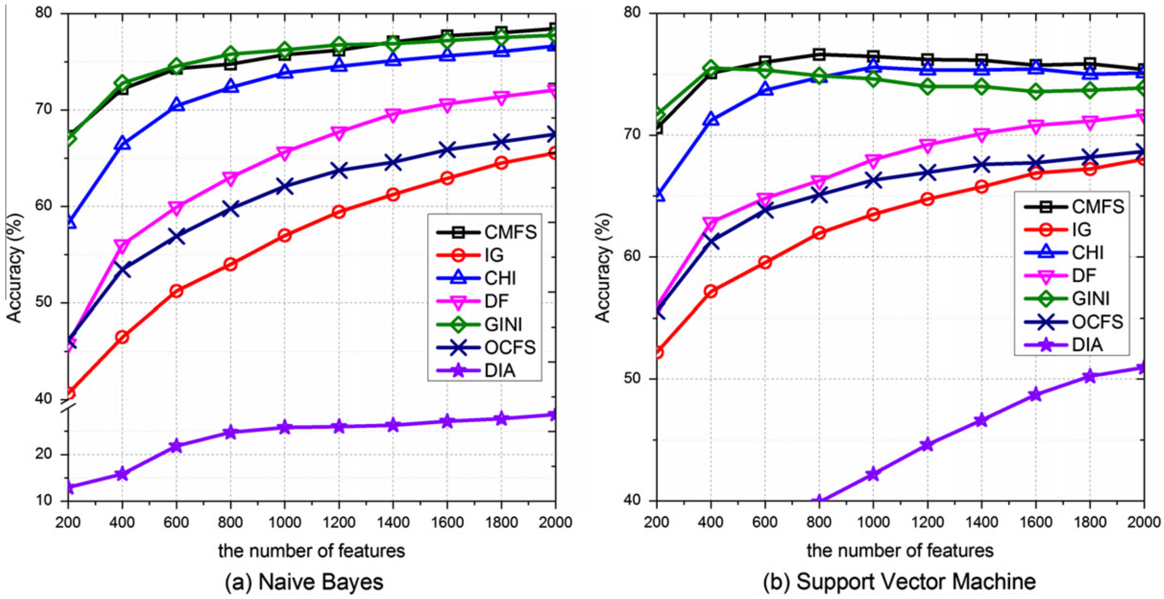
Table 3 and Table 4 show the micro F1 measure result when Naïve Bayes and Support Vector Machines are used on 20-newsgroups data set, respectively. It can be seen from Table 3 that the micro F1 performance of NB used CMFS on 20-newsgroups outperforms that based on the other algorithms when the number of the selected features is 200, 1400, 1600, 1800 or 2000. Moreover, the CMFS is only inferior to GINI when the number of the selected features is 400, 600, 800, 1000 or 1200. Table 4 indicates that the micro F1 performance of SVM used CMFS on 20-newsgroups is superior to that based on other feature selections when the number of selected features is greater than 400 and acquires the highest value (77.50%) when the number of the selected features is 800. Fig. 1 indicates the accuracy curves of Naïve Bayes and Support Vector Machines based on various feature selections, respectively. Fig. 1a shows that the accuracy curve of NB used CMFS on 20-newsgroups is higher than that based on other algorithms when the number of the selected features is 200, 1400, 1600, 1800 or 2000; the accuracy curve of SVM used CMFS on 20-newgroups is only lower than that based GINI when the number of the selected

Table 3
The micro F1 measure result using Naïve Bayes classifier on 20-newsgroups (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	**68.16**	73.05	75.11	75.68	76.70	77.13	**77.98**	**78.57**	**78.84**	**79.17**
IG	43.05	49.55	54.83	57.76	60.70	62.88	64.58	65.99	67.4	68.34
CHI	58.63	67.29	71.30	73.26	74.74	75.46	76.01	76.58	77.04	77.61
DF	48.19	59.23	63.47	66.37	68.58	70.32	71.90	72.78	73.49	74.07
GINI	67.81	**73.79**	**75.47**	**76.68**	**77.20**	**77.83**	77.91	78.21	78.49	78.74
OCFS	47.29	55.49	59.50	62.34	64.72	66.25	67.09	68.36	69.12	69.89
DIA	11.85	15.45	20.66	22.97	24.59	25.70	26.93	28.51	29.55	30.96

Table 4
The micro F1 measure result using SVM classifier on 20-newsgroups (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	73.41	76.91	**77.23**	**77.50**	**77.21**	**76.80**	**76.65**	**76.20**	**76.26**	**75.75**
IG	53.51	58.15	60.30	62.53	64.04	65.17	66.24	67.28	67.66	68.45
CHI	68.92	73.67	75.49	76.07	76.66	76.26	76.11	76.18	75.60	75.60
DF	57.00	63.57	65.28	66.69	68.37	69.57	70.51	71.23	71.55	72.06
GINI	**74.21**	**77.13**	76.28	75.58	75.18	74.50	74.42	73.99	74.07	74.26
OCFS	58.06	62.46	64.73	65.80	66.90	67.47	68.08	68.21	68.64	69.08
DIA	27.07	33.53	39.61	42.42	44.48	46.56	48.26	50.22	51.50	52.19



(a) Naive Bayes

(b) Support Vector Machine

Fig. 1. The accuracy performance curves of Naïve Bayes classifier and Support Vector Machines on 20-newsgroups, respectively.

features ranges from 400 to 1200. Fig. 1b indicates that the accuracy curve of SVM is higher than those based on other feature selections when the number of the selected features is greater than 400. The accuracy curves of NB based on CMFS ascends gradually with the increasing of the number of selected features, and the accuracy curve of SVM based on CMFS reaches the highest point (76.62%) when the number of the selected features is 800.

5.2. Experimental results on the Reuters-21578 dataset

Tables 5 and 6 show the micro F1 measure result when the Naïve Bayes and Support Vector Machines are used on Reuters-21578 data set, respectively. Table 5 indicates that the micro F1 performance of NB based on CMFS is superior to that based on the other feature selections except that the number of the selected features is 1200. It can be seen from Table 6 that the micro F1 performance of SVM used CMFS on Reuters-21578 outperforms that based on the other feature selections except that the number of selected features is 1400. The micro F1 of NB based on CMFS is the highest (66.84%) when the number of the selected features is 1800, and the micro F1 of SVM based on CMFS reaches the highest point (65.08%) when the number of the selected features is 200. Fig. 2a shows that the accuracy curve of NB based on CMFS is higher than that based on the other algorithms except that the number of the selected features is equal to 1200 or 1400. When the number of selected features is 1200 or 1400, the accuracy of NB based on CMFS is only inferior to that based GINI. Fig. 2b shows that the accuracy curve of SVM based on CMFS is also superior to that based on the other algorithms except that the number of the selected feature is 1200 or 1400.

5.3. Experimental results on the WebKB dataset

The micro F1 performance of Naïve Bayes and Support Vector Machines used on WebKB are listed in Tables 7 and 8, respectively. It can be seen from Table 7 that the micro F1 of NB based on CMFS is not the best and that based on GINI is the highest. Table 8 shows that the micro F1 of SVM based on CMFS outperforms that based on other algorithms when

Table 5
The micro F1 measure result using Naïve Bayes classifier on Reuters-21578 (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	**65.38**	**66.03**	**66.06**	**66.52**	**66.54**	66.38	**66.60**	**66.66**	**66.84**	**66.64**
IG	59.37	62.09	63.86	64.60	64.59	64.76	64.58	65.11	65.17	65.22
CHI	58.76	62.89	63.49	64.02	64.69	64.96	65.06	64.92	65.15	65.35
DF	58.05	62.41	63.27	64.09	64.97	64.99	65.31	65.36	65.41	65.51
GINI	64.08	65.13	65.73	65.82	66.14	**66.64**	66.53	66.16	66.03	65.78
OCFS	60.73	63.31	64.63	65.34	65.41	65.39	65.54	65.17	65.37	65.38
DIA	29.32	30.10	30.33	30.82	31.06	31.49	32.20	32.85	35.14	37.60

Table 6
The micro F1 measure result using SVM on Reuters-21578 (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	**65.08**	**63.73**	**63.52**	**63.52**	**63.16**	**63.14**	63.07	**63.06**	**63.06**	**62.97**
IG	62.19	61.31	61.73	62.19	62.20	62.66	62.59	62.59	62.58	62.88
CHI	63.08	62.85	63.05	63.12	62.72	62.96	62.88	62.93	62.84	62.73
DF	60.65	61.07	61.76	62.75	62.81	62.69	62.56	62.70	62.86	62.82
GINI	64.26	63.42	63.25	63.17	63.14	63.07	**63.19**	62.95	62.69	62.71
OCFS	62.95	62.58	62.44	62.15	62.50	62.36	62.28	62.56	62.68	62.87
DIA	41.96	45.06	46.90	48.87	50.27	51.05	50.76	51.85	52.58	53.04

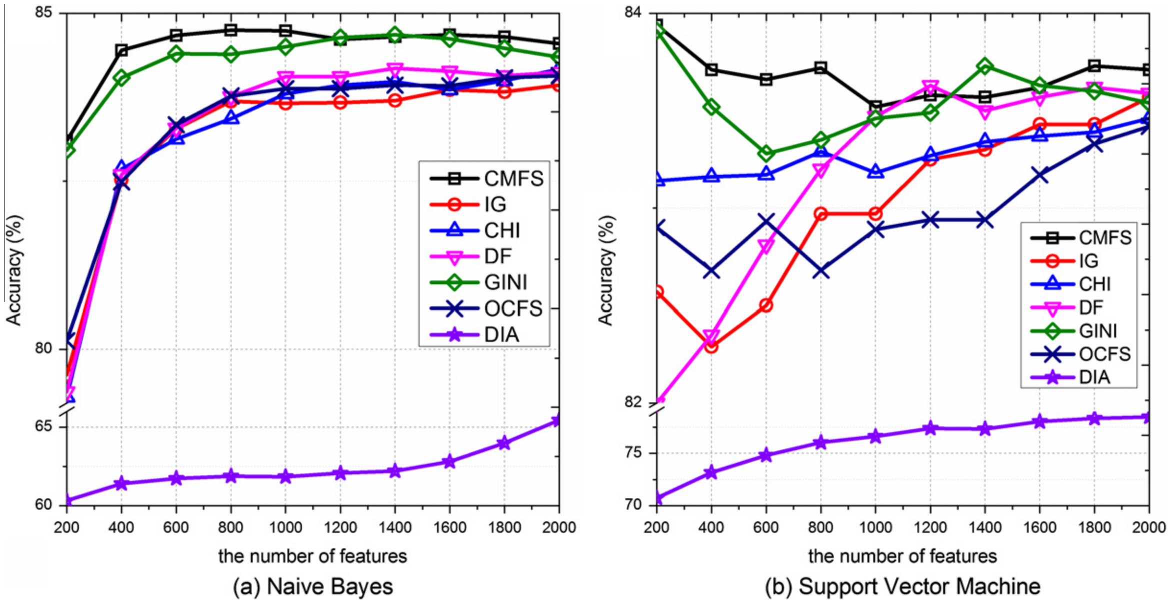


Fig. 2. The accuracy performance curves of Naïve Bayes classifier and Support Vector Machines on Reuters-21578, respectively.

the number of selected features is 1400 or 1600. The accuracy curves of NB and SVM used on WebKB are drawn in Fig. 3. Fig. 3a indicates that the accuracy curve of NB based on CMFS is lower than that based on GINI, CHI and OCFS, and higher than that based on DF and IG except that the number of the selected features is 800, 1800 or 2000. Fig. 3b shows that the accuracy curve of SVM based on CMFS is higher than that based on DF, IG and DIA, and exceeds the accuracy curve of SVM based on OCFS, GINI and CHI when the number of the selected features is 1400 or 1600.

6. Statistical analysis and discussions

6.1. Statistical analysis

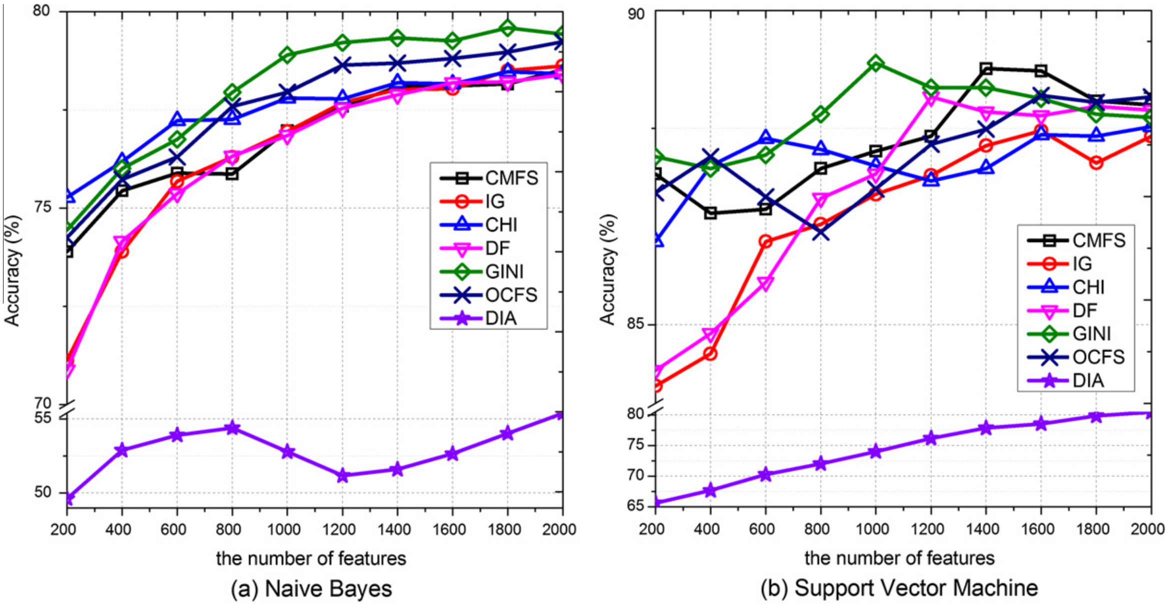
In order to compare the performance of the proposed method with the previous approaches, Friedman and Iman and Davenport (1980) tests are used in the statistical analysis. Friedman and Iman & Davenport tests (Demšar, 2006) are non-parametric tests. The null hypothesis of them is that all the algorithms are equivalent and so the ranks of all algorithms should be

Table 7
The micro F1 measure result using Naïve Bayes classifier on WebKB (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	68.78	72.14	72.77	73.22	74.71	75.62	76.45	76.81	77.02	77.33
IG	69.17	71.36	73.26	74.17	75.23	75.84	76.32	76.61	77.17	77.46
CHI	68.69	71.92	73.32	74.16	75.16	74.65	75.97	76.19	76.85	76.93
DF	67.37	71.04	72.91	73.74	74.82	75.82	76.29	76.87	76.98	77.23
GINI	68.45	72.13	**74.07**	**75.58**	**76.99**	**77.68**	**77.60**	**77.84**	**78.37**	**78.42**
OCFS	**70.71**	**72.96**	73.56	75.56	76.14	77.21	77.53	77.52	77.93	78.09
DIA	43.06	47.75	48.47	48.40	47.85	46.06	47.93	48.77	50.03	51.00

Table 8
The micro F1 measure result using SVM on WebKB (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

The number of features	200	400	600	800	1000	1200	1400	1600	1800	2000
CMFS	85.90	85.64	85.63	86.06	86.25	86.72	**87.53**	**87.63**	87.24	87.17
IG	82.97	83.52	85.22	85.66	86.02	86.10	86.65	86.80	86.52	86.77
CHI	84.13	85.88	**86.53**	86.52	86.20	86.08	86.40	86.89	86.89	86.89
DF	83.26	83.84	84.64	85.87	86.21	**87.27**	87.16	87.04	87.19	87.05
GINI	**86.01**	**86.17**	86.43	**86.81**	**87.48**	87.09	87.25	87.13	86.93	86.90
OCFS	85.47	86.04	86.05	85.57	86.07	86.66	86.74	87.37	**87.27**	**87.20**
DIA	59.80	62.66	66.9	69.13	72.32	74.84	76.47	76.95	78.54	79.03



(a) Naive Bayes

(b) Support Vector Machine

Fig. 3. The accuracy performance curves of Naïve Bayes classifier and Support Vector Machines on WebKB, respectively.

equal. If the null hypothesis of Friedman and Iman & Davenport tests is rejected, the post test (Bonferroni-Dunn test or Holm test) (García, Fernández, Luengo, & Herrera, 2009) can be used to detect significant differences among all the methods. Demšar (2006) detailed the computation of each test. In this paper, we compare the accuracy of seven feature selections using 30 data sets, which consist of the 10-fold cross validation on three data sets. Since the null hypothesis of Friedman and Iman & Davenport tests has been rejected, the Holm test, which can work with a control algorithm and compares it with remaining methods (García et al., 2009), is used to compare the CMFS with the other feature selection algorithms. Tables 9 and 10 show the Holm test table for $\alpha = 0.05$ when the Naïve Bayes and Support Vector Machines are used, respectively. It can be seen that the CMFS outperforms significantly the algorithms whose p-value is less than 0.025, and is comparable with the algorithms the p-value of which is greater than 0.025. Thus the CMFS outperforms significantly DIA, IG, CHI, DF, OCFS, and is comparable with GINI when Naïve Bayes classifier is used; the CMFS is superior to DIA, IG, DF and OCFS, and comparable with GINI and CHI when Support Vector Machines is used.

Table 9
Holm test table for $\alpha = 0.05$ when Naïve Bayes is used.

i	Algorithm	$z = (R_0 - R_i)/\mathrm{SE}$	p-value	Holm
6	CMFS vs. DIA	8.3367	7.6379E−17	0.0083
5	CMFS vs. IG	4.4223	9.7635E−6	0.01
4	CMFS vs. DF	3.8546	1.1591E−4	0.0125
3	CMFS vs. OCFS	2.8387	0.0045	0.0167
2	CMFS vs. CHI	2.3905	0.0168	0.025
1	CMFS vs. GINI	1.1355	0.2562	0.05

Table 10
Holm test table for $\alpha = 0.05$ when Support Vector Machines is used.

i	Algorithm	$z = (R_0 - R_i)/\mathrm{SE}$	p-value	Holm
6	CMFS vs. DIA	9.4124	4.8482E−21	0.0083
5	CMFS vs. IG	6.4841	8.9254E−11	0.01
4	CMFS vs. OCFS	4.9900	6.0354E−7	0.0125
3	CMFS vs. DF	4.2729	1.9291E−5	0.0167
2	CMFS vs. CHI	2.2411	0.0250	0.025
1	CMFS vs. GINI	0.8367	0.4029	0.05

6.2. Discussions

We take 10 features listed in Table 1 for consideration, and compared 5 features selected by DF, DIA and CMFS, respectively. It can be seen from Table 2 that the 5 features selected by CMFS consist of two best features selected by DF, two best features and “projects” selected by DIA; however, though “design”, “kits” and “sales” are deemed to contain more category information by DF, and “bank” and “designer” are deemed to be representative features for categorization by DIA, we think these features can not be regard as the best features for categorization based on theory of the CMFS. Since the CMFS accumulates the best features evaluated by DF and DIA, respectively, so the performance of the classifiers based on CMFS is superior to that based on the other feature selections.

Table 11 lists the top 10 features selected by various feature selection algorithms on 20-newgroups, Reuters-21578 and WebKB, respectively. It can be seen from Table 11 that some features are commonly selected by CMFS and the other feature selection algorithms. It is obvious that the classification abilities of common features are the same for various feature selection algorithms, so the rest of the features except for common features become the key for distinguishing the categorization abilities of various feature selection algorithms. In order to further validate the proposed method, 1000 features are selected by various feature selection algorithms on three datasets, respectively, and then the common features selected by CMFS and other feature selection algorithms are fed into the Naïve Bayes classifier. The performance of Naïve Bayes classifier based on the common features is compared with that based on the 1000 features selected by CMFS or the other feature selection algorithms. The results are listed in Table 12. It can be seen from Table 12 that the increments of CMFS are higher than other feature selection algorithms on 20-newgroups and Reuters-21578, so the rest of the features (except for common features) selected by CMFS can bring the more classification performance for Naïve Bayes on 20-newgroups and Reuters-21578.

All feature selection algorithms mentioned in this paper are filtering approaches, which measure the importance of a feature for the text categorization task based on the information theory or principles of statistics. Therefore, some selected features, such as “max”, “http”, “www”, “src”, etc., may be not very relevant to the categorization task in the view of the semantic. There exist two typical cases, some features are important for text categorization when they combined with other features; others are indeed irrelevant to the text categorization task and should be further filtered out.

It can be seen from Table 6 and Fig. 2 that the accuracy of SVM on Reuters-21578 is greater than 80%; however the micro F1 measure of SVM on Reuters-21578 is about 60%. We think the reason about this phenomenon is that the top 10 categories of Reutes-21578 is an imbalanced data set. Table 13 shows the number of documents in every category and the F1 measure of every category when various feature selections are used. It is worth noting that the number of documents in category “earn” is the most (3964) and the number of documents in category “corn” is only 237. The F1 measure of category “corn” and “wheat” is very low; namely, the performance of classification in category “corn” and “wheat” is very poor. Due to the number of documents in these two categories is very small, classifying incorrectly on minority category has little influence on accuracy of entire data set. So the accuracy metric in the imbalanced situation does not provide adequate information on a classifier’s functionality with respect to the type of classification required (He & Garcia, 2009). Moreover, Table 13 indicates that the improvement of F1 measure of CMFS on majority categories (acq, earn) is very limited, and even inferior to other methods; the F1 measure of the proposed method CMFS is increased in evidence on moderate and some category categories.

The performance of CMFS is, especially when Naïve Bayes is used, not optimal on WebKB. Through analyzing on the features selected by GINI and CMFS, we found that all documents in WebKB are web pages and the links in web page are the biggest noise source in the context of content-based classification. For example, the feature “uvic” only occurs 135 times in

Table 11
The top 10 features selected by various feature selections on three datasets, respectively.

	CMFS	IG	CHI	DF	GINI	OCFS	DIA
20-Newgroups	max	writes	aramis	writes	aramis	talk	mideast
	space	article	rutgers	article	rutgers	misc	turkish
	windows	don	approved	don	approved	politics	israeli
	christian	people	christian	people	christian	rutgers	arab
	graphics	cmu	politics	time	dod	approved	israel
	rutgers	time	mideast	cmu	forsale	christian	armenia
	god	talk	crypt	good	crypt	aramis	turks
	car	srv	clipper	apr	bike	srv	armenian
	turkish	xref	religion	xref	clipper	xref	arabs
	image	cantaloupe	talk	srv	space	cantaloupe	armenians
Reuters-21578	cts	mln	cts	mln	cts	cts	shr
	mln	dlrs	net	dlrs	net	net	qtr
	net	cts	shr	cts	shr	shr	revs
	loss	net	qtr	net	qtr	qtr	mths
	shr	pct	revs	loss	revs	oil	shrs
	oil	March	note	March	note	trade	div
	dlrs	year	loss	year	mln	wheat	avg
	trade	loss	profit	shr	loss	bank	dividend
	profit	billion	mths	pct	March	tonnes	qtly
	qtr	shr	trade	company	profit	company	cts
WebKB	www	http	professor	http	professor	professor	resume
	http	www	assignments	www	html	research	advisor
	src	computer	instructor	img	student	student	sports
	img	img	syllabus	src	page	assignments	friends
	gif	src	student	computer	home	instructor	favorite
	homework	gif	homework	gif	computer	university	music
	class	align	hours	align	university	class	china
	align	html	associate	html	text	interests	homes
	computer	content	research	content	type	hours	bookmarks
	page	page	class	page	content	fax	apt

Table 12
The growth pace of F1 measure (%). B denotes that the micro F1 performance of Naïve Bayes classifier when the common features selected by CMFS and the corresponding feature selection algorithm are used. The integers in parenthesis are the number of the common features; A indicates that the micro F1 performance of Naïve Bayes classifier when the features selected by CMFS are used. The decimals in parenthesis are the increment comparing A with B; C means that the micro F1 performance of Naïve Bayes classifier when the features selected by the corresponding feature selection algorithm are used. The decimals in parenthesis are the increment comparing C with B.

Datasets		IG	CHI	DF	GINI	OCFS	DIA
20-Newgroups	A	76.70(10.7)	76.70(−0.53)	76.70(10.67)	76.70(−0.32)	76.70(3.9)	76.70(54.9)
	B	66.00(333)	77.23(710)	66.03(320)	77.02(707)	72.80(511)	21.80(212)
	C	60.70(−5.3)	74.74(−2.49)	68.58(2.55)	77.20(0.18)	64.72(−8.08)	24.59(2.79)
Reuters-21578	A	66.54(1.88)	66.54(1.69)	66.54(2.27)	66.54(1.07)	66.54(1.07)	66.54(36.53)
	B	64.66(704)	64.85(733)	64.27(674)	65.47(830)	65.47(717)	30.01(401)
	C	64.59(−0.07)	64.69(−0.16)	64.97(0.7)	66.14(0.67)	65.41(−0.06)	31.06(1.05)
WebKB	A	74.71(1.51)	74.71(0.94)	74.71(1.49)	74.71(1.21)	74.71(1.09)	74.71(25.74)
	B	73.20(779)	73.77(657)	73.22(754)	73.50(768)	73.62(736)	48.97(305)
	C	75.23(2.03)	75.16(1.39)	74.82(1.6)	76.99(3.49)	76.14(2.52)	47.85(−1.12)

category “student”, and it is selected by the proposed method CMFS; however, the feature “uvic” is a part of a link “http://www.uvic.ca/” and does not contain any category information.

From the view of the theory of probability, the evaluation function of CMFS is similar to that of improved GINI index, except for the operation of squaring. We think there are two differences between CMFS and Improved GINI index. The first one is that the rationale of Improved GINI index is the measurement for the purity of attributes towards categorization (Shang et al., 2007), and the bigger the value of GINI of a term is, the better for categorization the term is; however, the theoretical principle of CMFS is that comprehensively evaluates the performance of a term for categorization based on both inter-category and intra-category. The second one is that the global significance of a term t_k in entire feature vector space with improved Gini index is calculated by summing the significance of the term t_k for every category up, but the proposed method CMFS computes the global performance of a term t_k in entire feature vector space through maximizing the significances of the term t_k for every category or weighted summing the significance of the term t_k for every category. Moreover, since the conditional probability that the feature t_k belongs to category c_i when the feature t_k occurs, $P(c_i|t_k)$ and the probability that

Table 13
The number of documents in each category in Reuters-21578 and F1 measure of each category when various feature selections are used (%). The bold values indicate the best performance of the classifier when various feature selection methods are used respectively.

Categories	Acq	Corn	Crude	Earn	Grain	Interest	Money-fx	Ship	Trade	Wheat
The number of documents	2369	237	578	3964	582	478	717	286	486	283
CMFS	**93.35**	**17.64**	**79.56**	96.94	49.05	**67.35**	**74.23**	**61.71**	**82.10**	14.10
IG	92.13	10.79	75.61	96.68	52.47	64.58	69.85	48.78	77.39	14.38
CHI	91.62	10.80	80.08	96.58	**57.69**	64.83	69.96	37.94	79.17	10.05
DF	92.01	16.25	74.70	96.95	46.37	64.73	70.07	41.32	77.86	15.31
GINI	93.14	15.66	78.25	**97.09**	46.13	65.98	69.65	46.40	76.39	**15.96**
OCFS	91.62	11.12	77.56	96.25	56.40	65.07	72.05	48.70	78.02	8.45

the feature t_k occurs in category c_i, $P(t_k \mid c_i)$ are greater than or equal to zero, the formula $P(t_k|c_i)^2P(c_i|t_k)^2$ is a monotonically increasing function. So the equation of the improved Gini index can be improved as follows:

$$\mathrm{Gini}(t_k) = \sum_i P(t_k|c_i)P(c_i|t_k) \tag{7}$$

It can be seen that the calculation of the significance of a term based on Eq. (7) and the proposed method CMFS is nearly the same except for the calculation of the global significance of the term. This may be the reason that the performance of NB and SVM based on CMFS is very close to that based on GINI.

7. Conclusion

In this paper, we have proposed a novel feature selection algorithm, named CMFS, by which the significance of a term both in inter-category and intra-category are comprehensively measured. The efficiency of the proposed measure CMFS was examined through the experiments of text categorization with NB and SVM classifier. The results, comparing with six well-known feature selection algorithms (Information Gain (IG), Improved Gini index (GINI), Chi square statistic (CHI), Document Frequency (DF), Orthogonal Centroid Feature Selection (OCFS) and DIA association factor (DIA)), show that the proposed method CMFS is significantly superior to DIA, IG, CHI, DF, OCFS when Naïve Bayes is used and significantly outperforms DIA, IG, DF, OCFS when Support Vector Machines is used. In the future, we will select features on imbalanced dataset and web dataset.

Acknowledgments

This research is supported by National Natural Science Foundation of China under Grant no. 60971089 and National Electronic Development Foundation of China under Grant no. 2009537.

References

Blum, A. L., & Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence, 97*, 245–271.

Chang, C. -C., & Lin, C. -J. (2001). LIBSVM: A library for support vector machines. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

Chen, J., Huang, H., Tian, S., & Qu, Y. (2009). Feature selection for text classification with Naïve Bayes. *Expert Systems with Applications, 36*, 5432–5435.

Chen, Z., & Lü, K. (2006). A preprocess algorithm of filtering irrelevant information based on the minimum class difference. *Knowledge-Based Systems, 19*, 422–429.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on, 13*, 21–27.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Drucker, H., Wu, D., & Vapnik, V. N. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural Networks, 10*, 1048–1054.

Fragoudis, D., Meretakis, D., & Likothanassis, S. (2005). Best terms: An efficient feature-selection algorithm for text categorization. *Knowledge and Information Systems, 8*, 16–33.

Fuhr, N., Hartmann, S., Lustig, G., Schwantner, M., Tzeras, K., Darmstadt, T. H., et al. (1991). AIR/X – A rule-based multistage indexing system for large subject fields. In: *Proceedings of the proceedings of RIAO* (pp. 606–623).

García, S., Fernández, A., Luengo, J., & Herrera, F. (2009). A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability. *Soft Computing – A Fusion of Foundations, Methodologies and Applications, 13*, 959–977.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*, 1263–1284.

Iman, R. L., & Davenport, J. M. (1980). Approximations of the critical region of the Friedman statistic. *Communications in Statistics, 18*, 571–579.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In C. Nédellec, & C. Rouveirol (Eds.), *Machine learning: ECML-98* (Vol. 1398, pp. 137–142). Berlin, Heidelberg: Springer.

John, G. H., Kohavi, R., & Pfleger, K. (1994). Irrelevant features and the subset selection problem. In Proceedings of the *machine learning: Proceedings of the eleventh international conference* (pp. 121–129). San Francisco, CA: Morgan Kaufmann Publishers.

Koller, D., & Sahami, M. (1997). Hierarchically classifying documents using very few words. Proceedings of the Proceedings of the Fourteenth International Conference on Machine Learning(ML-97), (pp. 170–178). Nashville,Tennessee.

Lang, K. (1995). NewsWeeder: Learning to filter netnews. In *Proceedings of the proceedings of ICML-95, 12th international conference on machine learning* (pp. 331–339).

McCallum, A., & Nigam, K. (1998). A comparison of event models for Naive Bayes text classification. In *Proceedings of the in: AAAI-98 workshop on learning for text categorization*.

Mengle, S. S. R., & Goharian, N. (2009). Ambiguity measure feature-selection algorithm. *Journal of the American Society for Information Science and Technology, 60*, 1037–1050.

Mladenic, D., & Grobelnik, M. (2003). Feature selection on hierarchy of web documents. *Decision Support Systems, 35*, 45–87.

Ogura, H., Amano, H., & Kondo, M. (2009). Feature selection with a measure of deviations from Poisson in text categorization. *Expert Systems with Applications, 36*, 6826–6832.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*, 1226–1238.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81–106.

Schneider, K.-M. (2004). A comparison of event models for Naive Bayes anti-spam E-mail filtering. *ACM Transactions on Asian Language Information Processing (TALIP), 3*, 243–269.

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

Shang, W., Huang, H., & Zhu, H. (2007). A novel feature selection algorithm for text categorization. *Expert Systems with Applications, 33*, 1–5.

Yan, J., Liu, N., Zhang, B., Yan, S., Chen, Z., & Cheng, Q., et al. (2005). OCFS: Optimal orthogonal centroid feature selection for text categorization. In *Proceedings of the proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 122–129). Salvador, Brazil: ACM.

Yang, J., Liu, Y., Liu, Z., Zhu, X., & Zhang, X. (2011). A new feature selection algorithm based on binomial hypothesis testing for spam filtering. *Knowledge-Based Systems, 24*, 904–914.

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *Proceedings of the in: Proceedings of the fourtheenth international conference on machine learning* (pp. 412–420). Morgan Kaufmann Publishers Inc.

Youn, E., & Jeong, M. K. (2009). Class dependent feature scaling method using Naive Bayes classifier for text datamining. *Pattern Recognition Letters, 30*, 477–485.