Text Clustering with Feature Selection by Using Statistical Data

Yanjun Li, Congnan Luo, and Soon M. Chung, *Member, IEEE*

***Abstract*— Feature selection is an important method for improving the efficiency and accuracy of text categorization algorithms by removing redundant and irrelevant terms from the corpus. In this paper, we propose a new supervised feature selection method, named CHIR, which is based on the χ^2 statistic and new statistical data that can measure the positive term-category dependency. We also propose a new text clustering algorithm TCFS, which stands for Text Clustering with Feature Selection. TCFS can incorporate CHIR to identify relevant features (i.e., terms) iteratively, and the clustering becomes a learning process. We compared TCFS and the k-means clustering algorithm in combination with different feature selection methods for various real data sets. Our experimental results show that TCFS with CHIR has better clustering accuracy in terms of the F-measure and the purity.**

***Index Terms*— Text clustering, text mining, χ^2 statistic, feature selection, performance analysis.**

I. Introduction

How to explore and utilize the huge amount of text documents is a major question in the areas of information retrieval and text mining. Document clustering is one of the most important text mining methods, which are developed to help users effectively navigate, summarize, and organize text documents. By organizing a large amount of documents into a number of meaningful clusters, document clustering can be used to browse a collection of documents or to organize the results returned by a search engine in response to a user's query. It can significantly improve the precision and recall in information retrieval systems [18], and it is an efficient way to find the nearest neighbors of a document [3]. The problem of document clustering is generally defined as follows: given a set of documents, we would like to partition them into a predetermined or an automatically derived number of clusters, such that the documents assigned to each cluster are more similar to each other than the documents assigned to different clusters. In other words, the documents in one cluster share the same topic, and the documents in different clusters represent different topics.

In most existing document clustering algorithms, documents are represented using the vector space model [18], which treats a document as a bag of words. A major characteristic of this representation is the high dimensionality of the feature space, which imposes a big challenge to the performance of clustering algorithms. They could not work efficiently in high dimensional feature spaces due to the inherent sparseness of the data [1]. Another problem is that not all features are important for document clustering. Some of the features may be redundant or irrelevant. Some may even misguide the clustering result, especially when there are more irrelevant features than relevant ones. In such case, selecting a subset of original features often leads to a better clustering performance [12]. Feature selection not only reduces the high dimensionality of the feature space, but also provides a better data understanding, which improves the clustering result. The selected feature set should contain sufficient or more reliable information about the original data set. For document clustering, this will be formulated into the problem of identifying the most informative words within a set of documents for clustering.

Feature selection has been widely used in supervised learning, such as text classification. It is reported that feature selection can improve the efficiency and accuracy of text classification algorithms by removing redundant and irrelevant terms from the corpus [23]. Traditional feature selection methods for classification are either supervised or unsupervised, depending on whether the class label information is required for each document. Those unsupervised feature selection methods, such as the ones using document frequency and term strength, can be easily applied to clustering [22]. But it is shown in [12] that supervised feature selection methods using the information gain [16] and the χ^2 statistic can improve the clustering performance better than unsupervised methods when the class labels of documents are available for the feature selection. However, supervised feature selection methods cannot be directly applied to document clustering because usually the required class label information is not available. In [12], an Iterative Feature Selection (IF) method is proposed, which utilizes the supervised feature selection to iteratively select features and perform text clustering.

In many previous text mining and information retrieval researches, the χ^2 term-category independence test has been widely used for the feature selection in a separate preprocessing step before text categorization [23]. By ranking their χ^2 statistic values, features that have strong dependency on the categories can be selected [12], [13]; and this method is denoted as CHI in this paper. Two variants of the χ^2 statistic have been proposed recently. In [14], *correlation coefficient* is proposed, which could be viewed as "one-sided" χ^2 statistic. Galavotti et al. [9] went further in this direction and proposed a simplified variant of the χ^2 statistic, which was called *GSS coefficient* in [19]. Feature selection methods based on these two variants of the χ^2 statistic were tested on improving the performance of text categorization in [9], [14].

In this research, we extended the χ^2 term-category independence test by introducing new statistical data that can measure whether the dependency between a term and a category is positive or negative. This new statistical data can describe the term-category dependency more accurately than the two variants of

Manuscript received September 26, 2006; revised September 30, 2007.

Yanjun Li is with the Department of Computer and Information Science, Fordham University, Bronx, NY 10458. Email:yli@fordham.edu.

Congnan Luo is with the Teradata Corporation, San Diego, CA 92127. Email:congnanluo@yahoo.com.

Soon M. Chung is with the Department of Computer Science and Engineering, Wright State University, Dayton, OH 45435. Email:soon.chung@wright.edu.

Digital Object Indentifier 10.1109/TKDE.2007.190740

the χ^2 statistic — correlation coefficient and GSS coefficient. We also developed a new supervised feature selection method, named CHIR, which is based on the χ^2 statistic and the new term-category dependency measure. Unlike CHI, CHIR selects features having strong positive dependency on the categories. In other words, CHIR keeps only the features relevant to the categories.

Furthermore, we explored CHIR in text clustering, and developed a new text clustering algorithm, named TCFS, which stands for Text Clustering with Feature Selection. Unlike the IF method [12], which performs text clustering and feature selection separately, TCFS integrates a supervised feature selection method, such as CHIR, into the text clustering process. Thus, TCFS is basically working as a learning process. While the information of the clusters is utilized to find more relevant features (i.e., terms), the quality of the clustering result is improved by reducing the weight of irrelevant features. As the TCFS algorithm converges, both a good clustering result and an informative feature subset are obtained.

Our experimental results with various real data sets demonstrated that the TCFS algorithm using the CHIR feature selection method performs better than k-means, k-means with the Term Strength (TS) feature selection method [12], the IF method, and TCFS with other supervised feature selection methods in terms of the accuracy of clustering results.

The rest of this paper is organized as follows. In Section 2, we describe the χ^2 term-category independence test and the feature selection method CHI. Then, we introduce a new term-category dependency measure and a new feature selection method CHIR. We also compare the new term-category dependency measure with the correlation coefficient and the GSS coefficient. In Section 3, we propose a high performance text clustering algorithm TCFS, which can adopt the feature section method CHIR without knowing the class information of the documents in advance. In Section 4, CHIR is compared with three feature selection methods, which are based on the χ^2 statistic and its two variants, in terms of the cluster cohesiveness. And the clustering accuracy of TCFS with CHIR is compared with those of other clustering and feature selection algorithms. Section 5 contains conclusions.

II. Feature Selection Based on the χ^2 Statistics

A. χ^2 Term-Category Independence Test

In text mining and information retrieval, we often use the χ^2 statistic to measure the degree of dependency between a term and a specific category. This can be done by comparing the observed co-occurrence frequencies in a 2-way contingency table with the frequencies expected when they are assumed to be independent. Suppose that a corpus contains n labeled documents, and they fall into m categories. After the stop words removal and the stemming, distinct terms are extracted from the corpus. We use an example to explain the χ^2 term-category independence test.

TABLE I
A 2×2 TERM-CATEGORY CONTINGENCY TABLE

	c	$\neg c$	$\sum$
w	40	80	120
$\neg w$	60	320	380
$\sum$	100	400	500

Example 1: To analyze the relationship between a term w and a category c, we create a two-way contingency table, shown as Table I. The row variable, term, has two possible values: $\{w, \neg w\}$. The column variable, category, may take either one in $\{c, \neg c\}$. Each cell at the position (i, j), where $i \in \{w, \neg w\}$ and $j \in \{c, \neg c\}$, contains the observed frequency, denoted by $O(i, j)$. For example, $O(w, c)$ is the number of documents which are in the category c and contain the term w, and $O(\neg w, \neg c)$ is the number of documents which neither belong to c nor contain w.

For the χ^2 term-category independence test, we consider the *null hypothesis* and the *alternative hypothesis*. The null hypothesis is that the two variables, term and category, are independent of each other. On the other hand, the alternative hypothesis is that there is some dependency between the two variables. To test the null hypothesis, we compare the observed frequency with the expected frequency calculated under the assumption that the null hypothesis is true. The expected frequency $E(i, j)$ can be calculated as:

$$E(i,j) = \frac{\sum_{a \in \{w, \neg w\}} O(a,j) \sum_{b \in \{c, \neg c\}} O(i,b)}{n} \tag{1}$$

The χ^2 statistic is defined as:

$$\chi^2_{w,c} = \sum_{i \in \{w, \neg w\}} \sum_{j \in \{c, \neg c\}} \frac{(O(i,j) - E(i,j))^2}{E(i,j)} \tag{2}$$

Equation 2 can be interpreted with the probabilities as follows:

$$\chi^2_{w,c} = \frac{n(p(w,c)p(\neg w, \neg c) - p(w, \neg c)p(\neg w, c))^2}{p(w)p(\neg w)p(\neg c)p(c)} \tag{3}$$

where $p(w, c)$ represents the probability that the documents in the category c contain the term w, $p(w)$ represents the probability that the documents in the corpus contain the term w, and $p(c)$ represents the probability that the documents in the corpus are in the category c, and so on. These probabilities are estimated by counting the occurrences of terms and categories in the corpus.

In Example 1, we get $E(w, c) = 24$, $E(w, \neg c) = 96$, $E(\neg w, c) = 76$, $E(\neg w, \neg c) = 304$, and $\chi^2_{w,c} = 17.61$. The degree of freedom is $(2 - 1) \times (2 - 1) = 1$ for our case. Looking up the table of the χ^2 distribution, we get the critical value $\chi^2_{0.001} = 10.83$ for the confidence level 0.1%. Since $\chi^2_{0.001}$ is much smaller than 17.61, we reject the null hypothesis. This can be explained as the divergence between the observed frequency and the expected frequency is statistically significant. That means, it is very unlikely that the divergence is caused only by the random sampling process. Thus, we believe there is some dependency between w and c; i.e., the distribution of the term w is related to the category c.

As shown in Equation 2, if the difference between the observed frequency and the expected frequency is bigger, then the χ^2 statistic becomes bigger, and the term is more informative for the category. This is the basic idea behind most previous researches on the feature selection for text categorization. The feature selection method CHI could be described as follows. For a corpus with m classes, the term-goodness of a term w is usually defined as either one of:

$$\chi^2_{avg}(w) = \sum_{j=1}^{m} p(c_j)\chi^2_{w,c_j} \tag{4}$$

$$\chi^2_{max}(w) = \max_j \{\chi^2_{w,c_j}\} \tag{5}$$

where $p(c_j)$ is the probability of the documents to be in the category c_j. Then, the terms whose term-goodness measure is lower than a certain threshold would be removed from the feature space. In other words, CHI selects terms having strong dependency on categories.

B. New Term-Category Dependency Measure $R_{w,c}$

In our research, we found the feature selection method CHI does not fully explore all the information provided by the χ^2 term-category independence test. We will use an example to point out where the problem is, and propose a new term-category dependency measure, denoted by $R_{w,c}$, to solve this problem.

TABLE II
ANOTHER 2×2 TERM-CATEGORY CONTINGENCY TABLE

	c	$\neg c$	$\sum$
w'	60	320	380
$\neg w'$	40	80	120
$\sum$	100	400	500

Example 2: Let's compare Table I and Table II. Using Equations 1 and 2, we can find both tables produce the same χ^2 statistic with $\chi^2_{w,c} = \chi^2_{w',c} = 17.61$. This is interesting because the two terms, w and w', actually have quite different distributions in c and $\neg c$.

We can see from Table I that there is positive dependency between w and c because $40/100 = 2/5$ of the documents in c contain w and $40/120 = 1/3$ of the documents containing w are in c. That means, w is a typical term in the category c, and w is relevant to c. On the other hand, as shown in Table II, it is not clear whether there is positive dependency between w' and c, because even though there are $60/100 = 3/5$ of the documents in c contain w', only $60/380 = 3/19$ of the documents containing w' are in c. In contrast, most documents in $\neg c$ contain w'. Therefore, it is difficult to believe that w' is relevant to c. In fact, we can claim that there is negative dependency between w' and c.

The second example shows that using only the χ^2 statistic might cause many errors in estimating how much a term is relevant to a category. To address this problem, we define our criterion for the relevancy of a term w to a category c as: the term w should have strong positive dependency on the category c. To evaluate whether the dependency between a term and a category is positive or negative, we introduce a new measure, $R_{w,c}$, defined as:

$$R_{w,c} = \frac{O(w,c)}{E(w,c)} \tag{6}$$

Equation 6 can be interpreted with the probabilities as follows:

$$R_{w,c} = \frac{p(w,c)p(\neg w,\neg c) - p(w,\neg c)p(\neg w,c)}{p(w)p(c)} + 1 \tag{7}$$

As $R_{w,c}$ is the ratio between $O(w,c)$ and $E(w,c)$, if there is no dependency between the term w and the category c (i.e., $\chi^2_{w,c}$ is not statistically significant), then $R_{w,c}$ should be close to 1. If there is positive dependency, then the observed frequency should be larger than the expected frequency, hence $R_{w,c}$ should be larger than 1. If there is negative dependency, $R_{w,c}$ should be smaller than 1.

From Equations 2 and 6, we can see the following relationship between $\chi^2_{w,c}$ and $R_{w,c}$: the farther $R_{w,c}$ is from 1, either negatively or positively, the bigger is its contribution to $\chi^2_{w,c}$. $\chi^2_{w,c}$ is a summary of the whole contingency table and just tells whether there is dependency between a term and a category in the distribution. But it cannot tell whether the dependency is positive or negative. On the other hand, $R_{w,c}$ tells the dependency more accurately. However, we still need to use $\chi^2_{w,c}$ to evaluate the dependency because our hypothesis test is based on the theoretical χ^2 distribution. By combining $\chi^2_{w,c}$ and $R_{w,c}$, we can provide better information about the dependency between a term and a category.

We estimate that the term w is relevant to the category c only when $\chi^2_{w,c}$ is statistically significant and $R_{w,c}$ is larger than 1. Using Equation 7, we get $R_{w,c} = 1.67$ for Table I and $R_{w',c} = 0.79$ for Table II. Based on our criteria, the term w has strong positive dependency on the category c and is relevant to c, while the term w' is irrelevant, which is a reasonable estimation.

Two variants of the χ^2 statistic have been proposed to address the same issue in a different approach. Ng et al. [14] suggested that a feature selection method should select the terms that belong to the relevant documents of a category and are indicative of the membership in the category. A variant of the χ^2 statistic, named *correlation coefficient*, was proposed in [14], and it can be viewed as "one-sided" χ^2 statistic. The *correlation coefficient* $\mathcal{C}$ for a term w and a category c is defined as:

$$\mathcal{C}_{w,c} = \frac{\sqrt{n}(p(w,c)p(\neg w,\neg c) - p(w,\neg c)p(\neg w,c))}{\sqrt{p(w)p(\neg w)p(\neg c)p(c)}} \tag{8}$$

The relation between $\mathcal{C}_{w,c}$ and $\chi^2_{w,c}$ is $\mathcal{C}^2_{w,c} = \chi^2_{w,c}$.

A simplified variant of the χ^2 statistic was proposed in [9]. It is based on the *correlation coefficient*, and called *GSS coefficient* in [19]. The *GSS coefficient* $s\chi^2$ for a term w and a category c is defined as:

$$s\chi^2_{w,c} = p(w,c)p(\neg w,\neg c) - p(w,\neg c)p(\neg w,c) \tag{9}$$

In order to emphasize the positive correlation between a term and a category, these two variants of the χ^2 statistic keep the second term in the numerator of $\chi^2_{w,c}$ in Equation 3 without the power of 2. Feature selection methods based on the *correlation coefficient* and the *GSS coefficient* are denoted by CC and SCHI, respectively, in this paper. Like CHI, for CC and SCHI methods, the term-goodness of a term in a corpus with m classes is defined as either the maximum or the average value of $\mathcal{C}_{w,c}$ and $s\chi^2_{w,c}$.

Comparing with $R_{w,c}$, the χ^2 statistic and its two variants are biased against the categories with small sizes when a term is uniformly distributed across multiple categories. Since $R_{w,c}$ is the ratio between $O_{w,c}$ and $E_{w,c}$, for the categories with different sizes, $R_{w,c}$ values are the same if the term has the same distribution in different categories. The following example explains the difference between $R_{w,c}$, χ^2 statistic, and its two variants in details.

TABLE III
7 DOCUMENTS IN 3 CATEGORIES WITH 5 TERMS

	c_1	c_2	c_3
w_1	d_7	d_1, d_2, d_3, d_5	
w_2	d_6, d_7	d_1, d_2, d_3, d_5	
w_3	d_6, d_7	d_1, d_2, d_5	
w_4		d_2, d_3	
w_5			d_4

TABLE IV
STATISTICAL VALUES FOR THE TERM w_2 WITH THE CATEGORIES c_1 AND c_2

	$\chi^2_{w_2,c_j}$	$\mathcal{C}_{w_2,c_j}$	$s\chi^2_{w_2,c_j}$	R_{w_2,c_j}
c_1	0.467	0.683	0.041	1.167
c_2	1.556	1.247	0.082	1.167

Example 3: Let's consider a set of seven labeled documents, $\{d_1, d_2, \ldots, d_7\}$, falling into three categories, $\{c_1, c_2, c_3\}$, as: $c_1 = \{d_6, d_7\}$, $c_2 = \{d_1, d_2, d_3, d_5\}$ and $c_3 = \{d_4\}$. There are total five distinct terms, $\{w_1, w_2, \ldots, w_5\}$, in the corpus, and the details are shown in Table III. Let's look at the distribution of the term w_2 in the categories c_1 and c_2: all the documents in the categories c_1 and c_2 contain the term w_2. Based on this observation, we can estimate that the term w_2 is equally relevant to the categories c_1 and c_2. For the term w_2, the values of χ^2 statistic, *correlation coefficient*, *GSS coefficient*, and $R_{w_2,c}$ are calculated and listed in Table IV. Since $\chi^2_{w_2,c_2}$ is larger than $\chi^2_{w_2,c_1}$, you may get the conclusion that w_2 is more relevant to c_2 than c_1, which is not supported by our observation. The statistical values of the *correlation coefficient* and the *GSS coefficient* also show the same trend. The reason for this problem is that the values of χ^2, $\mathcal{C}$ and $s\chi^2$ are affected by the sizes of the categories, and the size of c_1 is a half of the size of c_2. These three measures give more weight to the categories with bigger sizes, and it is not appropriate. On the other hand, our $R_{w,c}$ is not affected by the sizes of the categories, thereby R_{w_2,c_1} and R_{w_2,c_2} are the same (1.167). Based on this result, we can estimate that w_2 is equally relevant to c_1 and c_2, which is confirmed by the observation. This example shows that $R_{w,c}$ describes the term-category dependency more accurately than the χ^2 statistic and its two variants.

C. New Feature Selection Method CHIR

As we discussed in Section I, an appropriate feature selection method could improve the performance of text clustering by selecting the words that help to distinguish the documents into different clusters. First, let's study whether the feature selection method CHI is a good candidate for text clustering.

Recall that the feature selection method CHI uses the maximum or average χ^2 statistic value as the term-goodness measure to select the terms from the feature space. For Example 3 in Section II-B, the maximum and average χ^2 statistic values of five terms in the corpus are shown in Table V. By ranking the terms in descending order of their χ^2_{max} values, we can obtain a list as $(w_5, w_2, w_1, w_3, w_4)$. If we select the top three terms from this list, $\{w_5, w_2, w_1\}$ will be chosen. However, this selection is not good for text clustering. First, w_2 is ranked high and selected because it shows strong dependency on c_3, but in fact w_2 does not occur in any document in c_3. The strong dependency between w_2 and c_3, based on the χ^2 statistic, is negative dependency, which means that w_2 is irrelevant to c_3. Second, w_4 is not selected even though it is a good feature for distinguishing c_2, as we can see in Table III that w_4 occurs in all the documents of c_2.

If the average χ^2 statistic value of each term is used for ranking, the result is the same. Since w_2 occurs in all the documents of the two categories c_1 and c_2, with the contribution of large $\chi^2_{w_2,c_3}$ value, w_2 has a relatively large average χ^2 statistic value and is still ranked high. But a term like w_2, which is uniformly distributed across many categories, does not carry much useful information to distinguish the categories. Such redundant feature should not be kept in the feature space. Otherwise, the distinguishing power of other features like w_4 is depressed. This example shows that CHI method can remove the terms that are quite relevant to a category and keep the irrelevant and redundant terms. CHI method does not provide enough detail information about the relationship between the selected terms and the corresponding categories. To address this weakness, we propose a new feature selection method, named CHIR.

Based on the χ^2 statistic and $R_{w,c}$, we propose a new definition of the term-goodness of a term w in a corpus with m classes as:

$$r\chi^2(w) = \sum_{j=1}^{m} p(R_{w,c_j})\chi^2_{w,c_j} \text{ with } R_{w,c_j} > 1 \qquad (10)$$

where $p(R_{w,c_j})$ is the weight of χ^2_{w,c_j} in the corpus in terms of R_{w,c_j} and is defined as:

$$p(R_{w,c_j}) = \frac{R_{w,c_j}}{\sum_{j=1}^{m} R_{w,c_j}} \text{ with } R_{w,c_j} > 1 \qquad (11)$$

This new term-goodness measure, $r\chi^2(w)$, is the weighted sum of χ^2_{w,c_j} statistics when there is positive dependency between the term w and the category c_j, and a bigger $r\chi^2(w)$ value indicates that the term is more relevant. When the term w has negative dependency on the category c_j, its χ^2_{w,c_j} does not contribute to the calculation of $r\chi^2(w)$. Whether the term-category dependency is positive or negative is determined by $R_{w,c}$. According to the definition of $R_{w,c}$, when R_{w,c_j} is larger than 1, the dependency between w and c_j is positive; otherwise the dependency is negative.

When a term w has positive dependency on several categories, we believe that the stronger positive dependency between w and a category c should contribute more to the term-goodness of w, and its weight could be calculated in terms of $R_{w,c}$. The reason is that $R_{w,c}$ accurately measures the term-category dependency and is not affected by the sizes of categories.

For example, in Example 3, w_3 occurs in all the documents in c_1 and three out of the four documents in c_2, which indicates that the positive dependency between w_3 and c_1 shown by $\chi^2_{w_3,c_1}$ is stronger than that between w_3 and c_2. Thus, in calculating the term-goodness of w_3, $\chi^2_{w_3,c_1}$ should carry more weight than $\chi^2_{w_3,c_2}$. If the sizes of c_1 and c_2 are used to calculate the weight, the weight of $\chi^2_{w_3,c_1}$ will be less than that of $\chi^2_{w_3,c_2}$ because c_1 is smaller than c_2. On the other hand, when we use $R_{w,c}$ to calculate the weight, $p(R_{w_3,c_1})$ is larger than $p(R_{w_3,c_2})$ because R_{w_3,c_1} is larger than R_{w_3,c_2}. Another example is the term w_2 in Example 3. As we discussed before, w_2 has the same positive dependency on c_1 and c_2. In calculating the weight of $\chi^2_{w_2,c_j}$, R_{w_2,c_1} and R_{w_2,c_2} are better candidates than the sizes of the categories since they have the same value. These two examples show that, by using $R_{w,c}$ to calculate the weight, $r\chi^2(w)$ favors strong positive term-category dependency.

Our feature selection method CHIR uses $r\chi^2(w)$ to measure the term-goodness, and makes sure that the $r\chi^2$ statistic of each term represents only positive term-category dependency. The goal of this feature selection method is to find the terms that have strong positive dependency on certain categories in the corpus. In other words, CHIR selects the terms which are relevant to categories

TABLE V
THE χ^2 STATISTIC, $R_{w,c}$ AND $r\chi^2$ VALUES OF 5 TERMS

	c_1		c_2		c_3				
	$\chi^2_{w_i,c_1}$	R_{w_i,c_1}	$\chi^2_{w_i,c_2}$	R_{w_i,c_2}	$\chi^2_{w_i,c_3}$	R_{w_i,c_3}	χ^2_{max}	χ^2_{avg}	$r\chi^2$
w_1	0.630	0.700	3.733	1.400	2.917	0	3.733	2.730	3.733
w_2	0.467	1.167	1.556	1.167	7.000	0	7.000	2.022	1.012
w_3	1.120	1.400	0.058	1.050	2.917	0	2.917	0.770	0.665
w_4	1.120	0	2.100	1.750	0.467	0	2.100	1.587	2.100
w_5	0.467	0	1.556	0	7.000	7.000	7.000	2.022	7.000

and removes the irrelevant and redundant terms. The steps of CHIR to select q terms are as follows:

1) For each distinct term in the corpus, calculate its $r\chi^2$ statistic by using Equation 10.
2) Sort the terms in descending order of their $r\chi^2$ statistics.
3) Select the top q terms from the list.

For the term w_2 in Example 3, as shown in Table V, even though $\chi^2_{w_2,c_3} = 7$ is the largest among its χ^2 statistics, the corresponding $R_{w_2,c_3} = 0$ shows that w_2 has negative dependency on c_3. This is confirmed by the fact that w_2 never occurs in c_3 (see Table III). Thus, in our CHIR method, $r\chi^2(w_2)$ (1.012 in Table V) is obtained without the contribution of $\chi^2_{w_2,c_3}$. Similarly, for the term w_3, $r\chi^2(w_3)$ is 0.665 while its χ^2_{max} is 2.917. As a result of ranking the terms based on their $r\chi^2$ statistics, the new list of terms becomes $(w_5, w_1, w_4, w_3, w_2)$, and if we select the top three terms, $\{w_5, w_1, w_4\}$ will be selected. In Table III, we can see that these three terms are relevant to the corresponding categories, respectively, and they are more informative about the corpus than the three terms, $\{w_5, w_2, w_1\}$, selected by CHI. This example shows that CHIR selects a better feature subset than CHI.

III. TEXT CLUSTERING WITH FEATURE SELECTION (TCFS) ALGORITHM

A. Overview of Text Clustering

In most existing text clustering algorithms, text documents are represented by using the vector space model [18]. In this model, each document *d* is considered as a vector in the term-space and represented by the *term-frequency* (TF) vector:

$$d_{tf} = [tf_1, tf_2, \ldots, tf_h] \tag{12}$$

where tf_i is the frequency of the ith term in the document, and h is the dimension of the text database, which is the total number of unique terms. Normally there are several preprocessing steps, including the stop words removal and the stemming, on the documents. A widely used refinement to this model is to weight each term based on its *inverse document frequency* (IDF) [18] in the corpus. To account for the documents of different lengths, the length of each document vector is normalized to a unit length. In the rest of the paper, we assume that this normalized vector space model weighted by TF-IDF is used to represent documents during the clustering.

For the problem of clustering text documents, there are different criterion functions available. The most commonly used is the *cosine* function [18]. The *cosine* function measures the similarity between two documents as the correlation between the document vectors representing them. For two documents d_i and d_j, the similarity between them can be calculated as:

$$cosine(d_i, d_j) = \frac{d_i \bullet d_j}{\|d_i\|\|d_j\|} \tag{13}$$

where $\bullet$ represents the vector dot product and $\|d_i\|$ denotes the length of vector d_i. The *cosine* value is 1 when two documents are identical, and 0 if there is nothing in common between them. The larger *cosine* value indicates that these two documents share more terms and are more similar.

The k-means algorithm is very popular for solving the problem of clustering a data set into k clusters. If the data set contains n documents, $d_1, d_2, \ldots, d_n$, then the clustering is the optimization process of grouping them into k clusters so that the global criterion function

$$\sum_{j=1}^{k}\sum_{i=1}^{n} f(d_i, cen_j) \tag{14}$$

is either minimized or maximized. cen_j represents the centroid of cluster c_j, for $j = 1, \ldots, k$, and $f(d_i, cen_j)$ is the clustering criterion function for a document d_i and a centroid cen_j. When the *cosine* function is used, each document is assigned to the cluster with the most similar centroid, and the global criterion function is maximized as a result. This optimization process is known as an NP-complete problem [10], and the k-means algorithm was proposed to provide an approximate solution [11]. The steps of k-means are as follows:

1) Select k initial cluster centroids.
2) For each document of the whole data set, compute the clustering criterion function with each cluster centroid. Assign each document to its best choice. (*clustering step*)
3) Recalculate k centroids based on the documents assigned to them. (*updating step*)
4) Repeat Steps 2 and 3 until convergence.

B. Applying a Supervised Feature Selection Method to Text Clustering

It is challenging to apply supervised feature selection methods directly to text clustering because of the lack of the class label information of documents. However, it is not impossible to adopt the supervised feature selection in text clustering because the clusters obtained during the clustering process can provide valuable information for the feature selection. The well-known Expectation-Maximization (EM) algorithm [7] provides us a framework to combine the text clustering and supervised feature selection methods. Based on the EM algorithm, we propose a new text clustering algorithm, named Text Clustering with

Feature Selection (TCFS), which performs the clustering and the supervised feature selection alternately until convergence.

Recall that we defined the problem of text clustering as the grouping of documents with similar topics into a cluster. As we use the EM algorithm, we assume that each cluster of documents has a Gaussian distribution of terms. That means, a corpus with k clusters is considered as a mixture of k Gaussian distributions. Given the parameters and our Gaussian model, we maximize the likelihood of our data set. The maximum likelihood represents how well our Gaussian model fits the data set. In this case, the clustering criterion for the TCFS algorithm is the maximum likelihood, and a natural criterion for the feature selection also is the maximum likelihood.

For text clustering, the likelihood function $p(S|\theta)$, which represents the probability that a set S of n documents are grouped into k clusters when the parameter vector θ of the Gaussian model is given, can be written as:

$$p(S|\theta) = \prod_{i=1}^{n} \sum_{j=1}^{k} p(c_j|\theta) p(d_i|c_j, \theta) \tag{15}$$

where c_j is the jth cluster, $p(c_j|\theta)$ is the prior probability of the cluster c_j for given θ, and $p(d_i|c_j, \theta)$ is the prior probability of the document d_i in the cluster c_j for given θ. In the EM framework for text clustering, the terms in documents are assumed to be conditionally independent of each other, and the likelihood function can be rewritten as:

$$p(S|\theta) = \prod_{i=1}^{n} \sum_{j=1}^{k} (p(c_j|\theta) \prod_{w \in d_i} p(w|c_j, \theta)) \tag{16}$$

where $p(w|c_j, \theta)$ is the conditional probability of the term w in the cluster c_j.

As we discussed in Section I, not all the terms are equally relevant to the clusters, so $p(w|c_j, \theta)$ can be represented as:

$$p(w|c_j, \theta) = p_r(w|\theta) p_r(w|c_j, \theta) + (1 - p_r(w|\theta)) p_{ir}(w|c_j, \theta) \tag{17}$$

where $p_r(w|\theta)$ is the probability that the term w is relevant to the corpus for given θ, $p_r(w|c_j, \theta)$ is the probability of the term w in the cluster c_j for given θ when w is relevant, and $p_{ir}(w|c_j, \theta)$ is the probability of the term w in the cluster c_j for given θ when w is irrelevant. $p_r(w|\theta)$ is determined by performing a feature selection method. When a term w is selected, w is estimated to be relevant to the corpus and $p_r(w|\theta)$ is set to 1; otherwise, $p_r(w|\theta)$ is set to f, where f is a predetermined factor in the range of $[0, 1)$.

EM produces a sequence of estimates $\{\hat{\theta}(i)$ and $\hat{p_r}(i), i = 0, 1, 2, \ldots\}$ by using the following two steps:

1) Expectation step (E-step): $\hat{p_r}(i+1) = E(p_r|S, \hat{\theta}(i))$
2) Maximization step (M-step): $\hat{\theta}(i+1) = \arg \max_\theta p(S|\theta, \hat{p_r}(i))$

In fact, the E-step is to perform the supervised feature selection by calculating the expected feature relevancy for the current clustering result given, and the M-step is to re-cluster the data set in the new feature space.

Since the k-means clustering algorithm is considered as an extension of the EM framework for hard threshold cases [2], we can use k-means as the basis of our TCFS algorithm. In TCFS, a supervised feature selection method, such as CHIR, is integrated into the *updating step* of k-means, and the new *updating step* is considered as the E-step of TCFS. In each E-step, current cluster labels are treated as class labels, and CHIR is performed to estimate the relevancy of each term to the corpus, then the probability of the term relevancy, $p_r(w|\theta)$, is set to either 1 or f, where the range of f is $(0, 1)$ in TCFS. That means, if a term is selected based on the information obtained at the E-step, the term is estimated as relevant to the corpus and kept in the feature space. Otherwise, the term is estimated as irrelevant, and its weight is reduced by the factor of f; i.e., its new weight is calculated by multiplying the previous weight with f. After the feature selection, based on the documents assigned to them, k centroids are recalculated in the new feature space. In the M-step of TCFS, as in the *clustering step* of k-means, the documents in the corpus are re-clustered in the new feature space. The detail steps of our TCFS algorithm are as follows:

1) Perform a clustering algorithm, such as k-means, on the data set and get initial clusters.
2) Perform a supervised feature selection method, such as CHIR, on the data set by using the current clustering result as the class label information of the documents. The selected features (i.e., terms) remain untouched in the feature space, but the weight of each unselected feature is reduced by f, where f is a predetermined factor in the range of (0,1). Calculate k centroids in the new feature space. (E-step)
3) For each document in the corpus, compute the clustering criterion function with each cluster centroid in the new feature space. Assign each document to its best choice. (M-step)
4) Repeat Steps 2 and 3 until convergence.

The IF method proposed in [12] resolved the unavailability of the class label information by iteratively selecting features and performing clustering. IF also adopted the EM framework and the k-means algorithm, but there are two main differences between TCFS and IF. First, their M-steps are different. In the M-step of IF, the whole k-means algorithm is performed, which is independent of the feature selection method. The feature space does not change until the whole k-means algorithm is finished. On the other hand, in the M-step of TCFS, only the *clustering step* of the k-means algorithm is performed in the new feature space, which is obtained at each E-step, and it helps to produce an accurate final clustering.

Second, IF simply removes unselected features based on the relevancy score calculated in the E-step of every iteration. On the other hand, in the E-step of TCFS, we reduce the weight of unselected feature in the feature space. The class label information utilized by the supervised feature selection method is not the real (i.e., final) class label information of the documents. If the class labels are not correct at an iteration, some features may be mistakenly unselected. Once these unselected features are removed from the feature space at an early stage, they cannot be recovered later. For this reason, TCFS does not simply remove the unselected features at each iteration. With the convergence of EM iterations, we are getting closer to the real class label information, and eventually we could select the terms with high relevancy scores as the desired feature subset.

IV. Experimental Results

In this section, first the new feature selection method CHIR is compared with other feature selection methods CHI, CC (based on the *correlation coefficient*), and SCHI (based on the *GSS coefficient*) in terms of the cluster cohesiveness. Then, the

text clustering algorithm TCFS with different feature selection methods are compared with k-means, k-means with the Term Strength (TS), and the IF method. k-means is used as the basis for all the algorithms in the experiment. Since the performance of k-means is sensitive to the selection of initial centroids, for each test data set, we randomly selected 15 sets of k initial centroids. All the algorithms were tested with those sets of initial centroids, and the averages of results were used for the comparison. The experimental results show that TCFS with CHIR has the best clustering accuracy.

A. Data Sets

We used five test data sets extracted from two different types of text databases, which have been widely used by the researchers in the information retrieval area. Two data sets, denoted by CACM and MED, are extracted from the CACM and MEDLINE abstracts, respectively, which are included in the Classic database [4]. Additional three data sets, denoted by EXC, PEO and TOP, are from the EXCHANGES, PEOPLE and TOPICS category sets of the Reuters-21578 Distribution 1.0 [17].

Each document of the test data sets has been pre-classified into one unique class. But, this information was hidden during the clustering processes, and just used to evaluate the clustering accuracy of each clustering algorithm. Before the experiments, the stop words removal and the stemming were performed as preprocessing steps on the data sets. Table VI summarizes the characteristics of all the data sets used for our experiments.

B. Evaluation Methods

1) An Evaluation Method of Feature Selection: We used the cohesiveness of clusters to measure the performance of feature selection methods. The cohesiveness value of a cluster can be computed by using the weighed sum of the similarities between documents in the cluster as follows [20]:

$$\begin{aligned} Cohesiveness(c) &= \frac{1}{|c|^2}\sum_{d\in c, d'\in c} cosine(d', d) \\ &= \frac{1}{|c|}\sum_{d\in c} d \bullet \frac{1}{|c|}\sum_{d'\in c} d' \\ &= cen \bullet cen = \|cen\|^2 \end{aligned} \quad (18)$$

where c represents the cluster, cen is the centroid of the cluster, d and d' are documents in the cluster, and the $cosine$ function is used to measure the pairwise similarity between documents. From Equation 18, we can see that the square of the length of the centroid vector is the average pairwise similarity between two documents in the cluster. This also includes the similarity of each document with itself, which is one. When a feature selection method is applied to text documents, the cohesiveness value of each cluster may change. A good feature selection method should eliminate irrelevant features, while obtaining large cohesiveness values of the clusters.

2) Evaluation Methods of Text Clustering: We used the *F-measure* and the *purity* to evaluate the accuracy of the clustering algorithms.

The F-measure is a harmonic combination of the *precision* and *recall* values used in information retrieval [18]. Since our data sets were prepared as described in Section IV-A, each cluster obtained can be considered as the result of a query, whereas each pre-classified set of documents can be considered as the desired set of documents for that query. Thus, we can calculate the precision $P(i,j)$ and recall $R(i,j)$ of each cluster j for each class i.

If n_i is the number of the members of the class i, n_j is the number of the members of the cluster j, and n_{ij} is the number of the members of the class i in the cluster j, then $P(i,j)$ and $R(i,j)$ can be defined as:

$$P(i,j) = \frac{n_{ij}}{n_j} \quad (19)$$

$$R(i,j) = \frac{n_{ij}}{n_i} \quad (20)$$

The corresponding F-measure $F(i,j)$ is defined as:

$$F(i,j) = \frac{2 * P(i,j) * R(i,j)}{P(i,j) + R(i,j)} \quad (21)$$

Then, the F-measure for the whole clustering result is defined as:

$$F = \sum_i \frac{n_i}{n} \max_j(F(i,j)) \quad (22)$$

where n is the total number of documents in the data set. In general, the larger the F-measure is, the better the clustering result is [20].

The purity of a cluster represents the fraction of the cluster corresponding to the largest class of documents assigned to that cluster, thus the purity of the cluster j is defined as:

$$Purity(j) = \frac{1}{n_j}\max_i(n_{ij}) \quad (23)$$

The overall purity of the clustering result is a weighted sum of the purity values of the clusters:

$$Purity = \sum_j \frac{n_j}{n} Purity(j) \quad (24)$$

In general, the larger the purity value is, the better the clustering result is [24].

C. Comparison of Feature Selection Methods

In our experiment, feature selection methods CHIR, CHI, CC and SCHI were evaluated. In order to eliminate the effect of text clustering algorithms on the experiment, we ran the four feature selection methods on labeled text documents. Each class of labeled documents was treated as a cluster, and the cohesiveness values of each class after the feature selection were compared. For CHI, CC and SCHI, we used $\chi^2_{max}(w)$, $\mathcal{C}_{max}(w)$ and $s\chi^2_{max}(w)$ as the term-goodness measure, respectively, because the maximum values of χ^2 and $s\chi^2$ statistics were reported to perform better than their average values [9], [23].

In our experiment, the percentage of the selected features (i.e., terms) was varied from 5% to 90%. At each round of feature selection, the unselected terms were simply removed from the feature space, then the document vectors were re-normalized. For comparison, the cohesiveness value of each class was calculated and recorded.

We evaluated four feature selection methods (CHIR, CHI, CC and SCHI) on 43 classes of the CACM data set with different percentages of feature selection. The cohesiveness values of the classes were compared, and a part of the results is shown in Figures 1 and 2. With fewer terms left in the feature space, the cohesiveness value of the cluster increases because sparse features are removed and documents become more similar to each other. When the feature selection method selects an appropriate feature

TABLE VI
SUMMARY OF DATA SETS

Data Set	Num. of Doc.	Num. of Classes	Min. Class Size	Max. Class Size	Num. of Unique Terms	Avg. Doc. Length	Avg. Pairwise Similarity by cosine
CACM	842	43	11	51	3,225	59	0.03
MED	287	9	26	39	4,255	77	0.02
EXC	334	7	28	97	3,258	67	0.03
PEO	694	15	11	143	5,046	102	0.04
TOP	2279	7	23	750	10,719	113	0.03

subset, which represents the cluster better than other subsets, the cohesiveness value is larger. For example, for the class c_1 of the CACM data set, when CHIR is performed with 20% of the features selected, the cohesiveness value of this class is 0.24. When CHI, CC and SCHI are performed, the cohesiveness values of c_1 are 0.201, 0.225 and 0.172, respectively. This result suggests that CHIR removes irrelevant features better than CHI, CC and SCHI when 20% of the features are selected.

Our experimental results indicate that CHIR consistently outperforms other three methods in terms of increasing the cohesiveness values of the clusters. The performance of CC and CHI are very close in some cases, which tells us that identifying only the positive term-category dependency is not sufficient. The term-goodness measure should also describe the dependency accurately. SCHI does not perform well in most cases because the definition of $s\chi^2$ (see Equation 9) does not contain sufficient information about the χ^2 term-category independence test.

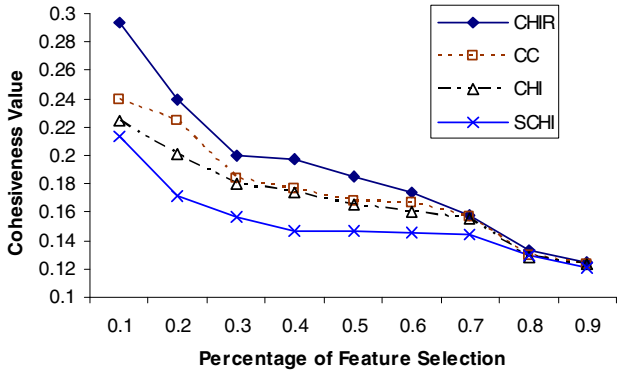

Fig. 1. Cohesiveness values of the class c_1 of the CACM data set

D. Comparison of Text Clustering Algorithms

In [12], supervised and unsupervised feature selection methods were evaluated in terms of improving the clustering performance by conducting experiments in the case that the class labels of documents are available for the feature selection. As a preprocessing step of text clustering, the Term Strength (TS) feature selection method was reported as the best among the unsupervised feature selection methods evaluated in [12].

TS was originally proposed and evaluated for the vocabulary reduction in text retrieval [21], and later applied to text categorization [22]. It is computed based on the conditional probability that

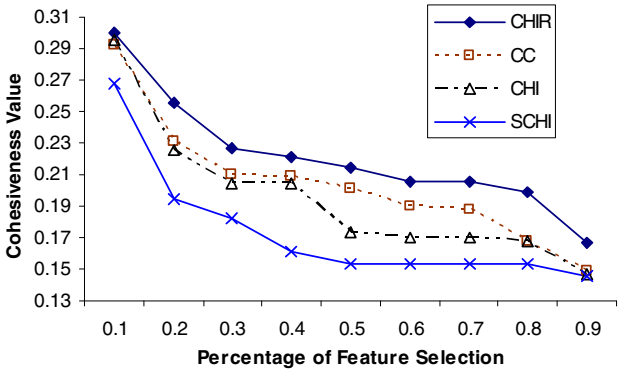

Fig. 2. Cohesiveness values of the class c_2 of the CACM data set

a term occurs in the second one of a pair of related documents, given that it occurs in the first one:

$$TS(w) = p(w \in d_j \mid w \in d_i) \text{ with } similarity(d_i, d_j) \geq \delta \quad (25)$$

$$\approx \frac{\text{\# of pairs in which } w \text{ occurs in both documents}}{\text{\# of pairs in which } w \text{ occurs in the first document}} \quad (26)$$

where δ is the parameter to determine the related document pairs. Since we need to calculate the similarity of each document pair, the time complexity of calculating TS is quadratic of the number of documents. As the class label information is not required, TS can be used for the term reduction in text clustering. In this case, terms are ordered in descending order of their TS values, and then a certain percentage of them are selected from the top to be used for clustering.

Our experiment in this section has two parts. First, we compared the clustering accuracy of k-means, k-means with TS, and TCFS with three different feature selection methods. When k-means was combined with TS, δ was set to 0.1% and the feature selection was performed first as a preprocessing step, then k-means was applied to the data set in the new feature space. The percentage of feature selection was varied in the range of [5%, 90%] for all feature selection methods.

When we performed TCFS on the data sets, at each iteration, a certain percentage of features were selected based on the supervised feature selection method chosen — CHIR, CHI or CC. We did not test TCFS with SCHI because it was reported in [9] that $s\chi^2_{max}(w)$ improves the performance of the text categorization only when extremely aggressive feature selection is applied. For CHI and CC, $\chi^2_{max}(w)$ and $\mathcal{C}_{max}(w)$ were chosen as

the term-goodness measure, respectively. As described in Section III, the relevancy of each term to the clusters is estimated based on the information obtained at each iteration. The probability of the term relevancy is set to either 1 or f, where f is a predetermined factor in the range of (0,1). At each iteration, the weight of each irrelevant term is reduced by f in the feature space. In our experiments, we set f as 0.5.

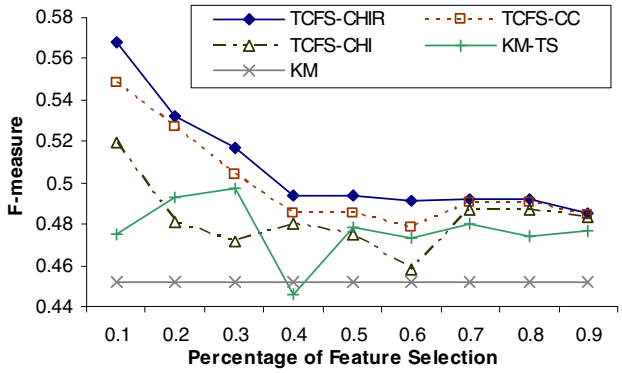

Fig. 3. F-measure of the clusters of the EXC data set

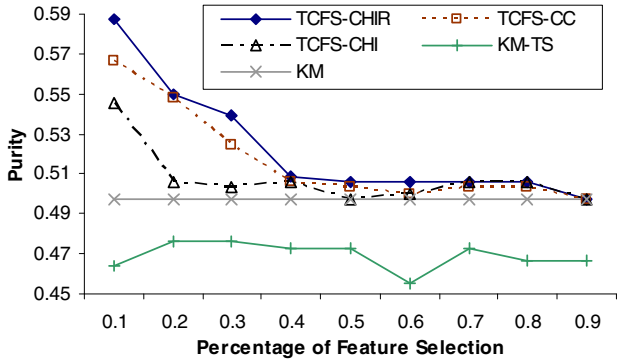

Fig. 4. Purity value of the clusters of the EXC data set

Figures 3 and 4 show the results of running five clustering algorithms on the EXC data set. Tables VII and VIII show the F-measure and purity values of the clusters obtained by running five clustering algorithms on different data sets when 25% of the terms were selected. A few conclusions can be drawn from the experimental results:

- Feature selection methods can improve the performance of text clustering as more irrelevant or redundant terms are removed.
- TCFS with a supervised feature selection method, such as CHIR, CHI or CC, can achieve a better F-measure than k-means with TS. The results suggest that performing a supervised feature selection method based on the information of clusters obtained during the clustering process can improve the clustering accuracy.
- k-means with TS does not consistently outperform k-means as the percentage of feature selection is varied. For example, when 40% of the terms of the EXC data set are selected by TS, the F-measure is 0.446, which is even lower than the case of simply performing k-means. The purity values of the clustering results obtained by performing k-means with TS on the EXC data set are consistently lower than those of performing k-means. This result shows that TS does not always select an appropriate feature subset. In certain cases, TS removes some relevant terms, while keeping some irrelevant ones.
- TCFS with CHIR outperforms all other clustering algorithms in terms of the F-measure and the purity value for different test data sets.

Second, we compared TCFS with the IF method. For these two algorithms, we applied CHIR and CHI as the supervised feature selection methods. The F-measure of the clusters of the EXC and PEO data sets obtained by running TCFS and the IF method are shown in Figures 5 and 6. Due to limited figure size, we show the percentage of feature selection only in the range of [50%, 90%]. Both the whole process of TCFS and the IF method with one iteration of k-means involve two complete rounds of the k-means algorithm, so they are comparable with each other. The experimental results suggest that the performance of TCFS is better than that of IF (with one iteration of k-means) in most cases. The reason is because IF performs the feature selection separately from k-means.

The main weakness of the IF method is: it does not always improve the clustering performance with more iterations. For example, when 70% of the terms of the PEO data set are selected at each iteration, the F-measure of the clustering result drops from 0.597 to 0.575 with two more iterations (see Fig. 6). Since the cluster labels used by the supervised feature selection are not real class labels, simply removing unselected features from the feature space is not desirable. In IF, once a feature is unselected, it is removed, so there is no chance to recover it later. In our TCFS algorithm, we keep the unselected features but reduce their weight in the feature space. This approach is safer because a mistake at early stage could be corrected later as the unselected features are still kept in the feature space. This helps to obtain a better clustering result as the algorithm converges.

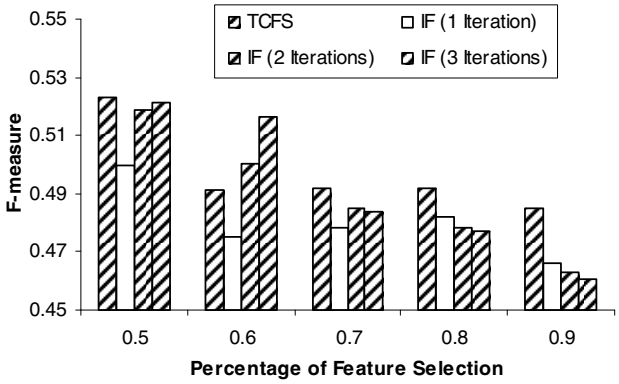

Fig. 5. F-measure of the clusters of the EXC data set (CHIR is adopted for TCFS and IF)

V. Conclusions

In this paper, we introduced a new term-category dependency measure, denoted by $R_{w,c}$, which can tell whether the dependency is positive or negative and describe the dependency more accurately. Based on the χ^2 statistic and $R_{w,c}$, we proposed a new supervised feature selection method CHIR. CHIR selects the terms that are relevant to the categories by utilizing the known class label information. CHIR can be used for text categorization,

TABLE VII
F-MEASURE OF THE CLUSTERS (25% FEATURE SELECTION)

Data Set	KM	KM with TS	TCFS with CHI	TCFS with CC	TCFS with CHIR
CACM	0.429	0.428	0.478	0.478	0.481
MED	0.569	0.702	0.737	0.747	0.753
EXC	0.452	0.495	0.477	0.506	0.522
PEO	0.582	0.604	0.607	0.608	0.613
TOP	0.621	0.561	0.640	0.642	0.667

TABLE VIII
PURITY VALUES OF THE CLUSTERS (25% FEATURE SELECTION)

Data Set	KM	KM with TS	TCFS with CHI	TCFS with CC	TCFS with CHIR
CACM	0.532	0.524	0.597	0.601	0.607
MED	0.606	0.672	0.746	0.750	0.756
EXC	0.497	0.476	0.506	0.529	0.542
PEO	0.586	0.608	0.600	0.600	0.611
TOP	0.775	0.709	0.786	0.787	0.790

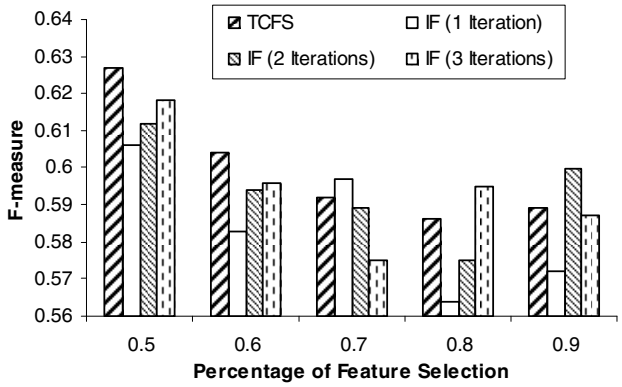

Fig. 6. F-measure of the clusters of the PEO data set (CHI is adopted for TCFS and IF)

text summarization, and ontology creation. Our experimental results using real data sets show that CHIR generates more cohesive clusters than the existing feature selection methods based on the χ^2 statistic, *correlation coefficient*, and *GSS coefficient*.

We also proposed a new text clustering algorithm TCFS that performs a supervised feature selection during the clustering process. The cluster label information obtained during the clustering process is utilized as the known class label information for the feature selection. The selected features improve the quality of clustering iteratively, and as the clustering process converges, the clustering result has higher accuracy.

TCFS with CHIR has been compared with other clustering and feature selection algorithms, such as k-means, k-means with the Term Strength (TS) feature selection method, the IF method, and TCFS with other feature selection methods. Our experimental results show that TCFS with CHIR has better performance than other algorithms in terms of the clustering accuracy for different test data sets.

ACKNOWLEDGMENT

This research was supported in part by AFRL/Wright Brothers Institute (WBI).

REFERENCES

[1] C. C. Aggrawal and P. S. Yu, "Finding Generalized Projected Clusters in High Dimensional Spaces," Proc. of ACM SIGMOD Int'l Conf. on Management of Data, pp. 70–81, 2000.
[2] L. Bottou and Y. Bengio, "Convergence Properties of the K-means Algorithms," Advances in Neural Information Processing Systems 7, pp. 585–592, 1994.
[3] C. Buckley and A. F. Lewit, "Optimizations of Inverted Vector Searches," Proc. of Annual ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 97–110, 1985.
[4] Classic data set, available at ftp://ftp.cs.cornell.edu/pub/smart/.
[5] M. Dash and H. Liu, "Feature Selection for Classification," Intelligent Data Analysis, vol. 1, no. 3, pp. 131–156, 1997.
[6] M. Dash and H. Liu, "Feature Selection for Clustering," Proc. of Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), pp. 110–121, 2000.
[7] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," Journal of the Royal Statistical Society, vol. 39, no. 1, pp. 1–38, 1977.
[8] G. Forman, "Feature Selection: We've Barely Scratched the Surface," IEEE Intelligent Systems, November, 2005.
[9] L. Galavotti, F. Sebastiani, and M. Simi, "Experiments on the Use of Feature Selection and Negative Evidence in Automated Text Categorization," Proc. of the 4th European Conf. on Research and Advanced Technology for Digital Libraries, pp. 59–68, 2000.
[10] M. R. Garey, D. S. Johnson, and H. S. Witsenhausen, "Complexity of the Generalized Lloyd-Max Problem," IEEE Trans. on Information Theory, vol. 28, no. 2, pp. 255–256, 1982.
[11] J. A. Hartigan, *Clustering Algorithms*, John Wiley & Sons, 1975.
[12] T. Liu, S. Liu, Z. Chen, and W. Ma, "An Evaluation on Feature Selection for Text Clustering," Proc. of Int'l Conf. on Machine Learning, 2003.
[13] C. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
[14] H. T. Ng, W. B. Goh, and K. L. Low, "Feature Selection, Perception Learning, and a Usability Case Study for Text Categorization," Proc. of the 20th ACM Int'l Conf. on Research and Development in Information Retrieval, pp. 67–73, 1997.
[15] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant Features and the Subset Selection Problem," Proc. of Int'l Conf. on Machine Learning, pp. 121–129, 1994.

[16] J. R. Quinlan, "Induction of Decision Trees," Machine Learning, vol. 1, pp. 81–106, 1986.
[17] Reuters-21578 Distribution 1.0, available at http://www.daviddlewis.com/resources/testcollections/reuters21578.
[18] C. J. van Rijsbergen, *Information Retrieval*, 2nd edition, Butterworth, London, 1979.
[19] F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1–47, 2002.
[20] M. Steinbach, G. Karypis, and V. Kumar, "A Comparison of Document Clustering Techniques," Proc. KDD Workshop on Text Mining, 2000.
[21] W. J. Wilbur and K. Sirotkin, "The Automatic Identification of Stop Words," Journal of Information Science, vol.18, no. 1, pp. 45–55, 1992.
[22] Y. Yang, "Noise Reduction in a Statistical Approach to Text Categorization," Proc. of Annual ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 256–263, 1995.
[23] Y. Yang and J. O. Pedersen, "A Comparative Study on Feature Selection in Text Categorization," Proc. of Int'l Conf. on Machine Learning, pp. 412–420, 1997.
[24] Y. Zhao and G. Karypis, "Empirical and Theoretical Comparisons of Selected Criterion Functions for Document Clustering," *Machine Learning*, vol. 55. no. 3, pp. 311–331, 2004.

PLACE PHOTO HERE

Yanjun Li received the B.S. degree in Economics from the University of International Business and Economics, Beijing, P.R. China, in 1993, the B.S. degree in Computer Science from Franklin University, Columbus, Ohio, in 2001, the M.S. degree in Computer Science and the Ph.D. degree in Computer Science and Engineering from Wright State University, Dayton, Ohio, in 2003 and 2007, respectively. She is currently an assistant professor in the Department of Computer and Information Science at Fordham University, Bronx, New York. Her research interests include data mining and knowledge discovery, text mining, ontology, information retrieval, bioinformatics, and parallel and distributed computing.

PLACE PHOTO HERE

Congnan Luo received the B.E. degree in Computer Science from Tsinghua University, P.R. China, in 1997, the M.S. degree in Computer Science from the Institute of Software, Chinese Academy of Sciences, Beijing, P.R. China, in 2000, and the Ph.D. degree in Computer Science and Engineering from Wright State University, Dayton, Ohio, in 2006. Currently he is a technical staff at the Teradata Corporation in San Diego, CA, and his research interests include data mining, machine learning, and databases.

PLACE PHOTO HERE

Soon M. Chung received the B.S. degree in Electronic Engineering from Seoul National University, Korea, in 1979, the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology, Korea, in 1981, and the Ph.D. degree in Computer Engineering from Syracuse University, Syracuse, New York, in 1990. He is currently a professor in the Department of Computer Science and Engineering at Wright State University, Dayton, Ohio. His research interests include database, data mining, Grid computing, text mining, XML, and parallel and distributed processing.