Clustering of Words Based on Relative Contribution for Text Categorization

Jie-Ming Yang, Zhi-Ying Liu, Zhao-Yang Qu

Abstract **—Term clustering tries to group words based on the similarity criterion between words, so that the groups can be used as the dimensions of the vector space in the text categorization. We proposed two new similarity criterions, which consider the relative contribution of one feature for one category relative to other categories and the difference between relative contributions of two features. We used the proposed methods in hierarchical clustering algorithm, and generated a compact and efficient representation of documents. The proposed methods are evaluated on three benchmark corpora (20-newgroups, reuters-21578 and industry sector), combined with two classification algorithms (Support Vector Machines, K-Nearest Neighbor), and compared with three similarity measures (weighted average KL divergence, City-block, Euclidean). The experimental results indicated that the performance of the proposed methods are comparative with the other methods when Support Vector Machines is used; the proposed methods significantly outperform Euclidean and City-block and achieve comparative performance with weighted average KL divergence when K-Nearest Neighbor classifier is used.**

Index Terms**—term clustering, similarity measure, text categorization, relative contribution**

I. Introduction

Automatic document categorization, which assigns the predefined categories to a new text document, is an important tool for people to organize the vast amount of digital data in the Internet [1]. The raw documents cannot be directly fed into a classifier, so they must be transformed into a uniform representation form based on their content [2]. Many experiments show that the document representation approach based on bag of words (BOW) is a sophisticated one [2]. A major characteristic of text categorization is the high dimensionality of the feature vector space, which can be tens or hundreds of thousands of terms for even a moderated size dataset [3, 4]. It is a big hurdle in applying many sophisticated learning algorithms to the text categorization [5]. Another major characteristic of text categorization are the high level of feature redundancy, feature irrelevance [4] and sparseness problem [6]. These characteristics not only hinder the classification process and hurt the performance of the classifier but also bring about over-fitting. Because of this, the dimensionality reduction is used to reduce the size of the feature vector space [2].

The term clustering is one of the dimensionality reduction methods [2, 7]. It can create a new, reduced-size feature space by grouping words with high similarity [1], and many words can be replaced with the centroid or representative feature of the corresponding word cluster. There are three key benefits of the term clustering: (1) the features that have correlations on the class labels assigned to documents are considered as a new feature in the reduced-size feature space. (2) The term clustering can result in higher classification accuracy. (3) The term clustering can provide a good solution to the sparseness problem and generate extremely compact representations [1, 6, 7].

The crucial stage of term clustering is how to measure the similarity of the terms [8]. The measurement of the similarity between two elements is essential to the most clustering procedures. Based on the similarity, one can decide which clusters should be combined or where a cluster should be split. McGill [9] listed more than 60 different similarity functions. The quality of clustering depends on whether the similarity metric is appropriate or not. There are many similarity measurements which are widely used in

This research is supported by National Natural Science Foundation of China under Grant no. 51077010.

Jie-Ming Yang is with College of Information Engineering, Northeast Dianli University, Jilin, Jilin, China (e-mail: yjmlzy@gmail.com).

Zhi-Ying Liu is with College of Information Engineering, Northeast Dianli University, Jilin, Jilin, China (e-mail: yjmlzy@163.com).

Zhao-Yang Qu is with College of Information Engineering, Northeast Dianli University, Jilin, Jilin, China (e-mail: qzywww@mail.nedu.edu.cn)

clustering, such as weighted average KL divergence [1], Euclidean metric [10, 11], City-block metric [12].

However, not all proximity measures are applicable in each environment [13]. Two new similarity measures are proposed for term clustering in this paper. The contribution of a feature to one category is represented by the sum of term frequency occurred in it. The relative contribution of a feature occurring in one category relative to other categories is considered. If the relative contributions of one feature occurring in each category are same as that of the other one, there exists the highest similarity between these two features in terms of contribution to categorization. The difference between the relative contributions of two features is regarded as a new similarity measure. To evaluate the proposed method, we used two classification algorithms, Support Vector Machines (SVM) and K-nearest neighbors (KNN) on three benchmark text corpora (20-newsgroups, Reuters-21578 and Industry Sector) and compared it with three well-known similarity measures (weighted average KL divergence, City-block, Euclidean). The experiments show that the performance of the proposed methods are comparative with the others when Support Vector Machines is used; the proposed methods significantly outperform Euclidean and City-block and achieve comparative performance with weighted average KL divergence when K-Nearest Neighbor classifier is used.

II. Related Work

Much study has been devoted to word clustering for text categorization in recent years. In this section we overview the results which are most relevant to our study.

Word clustering is firstly investigated and used in text categorization by Lewis [14]. Lewis used reciprocal nearest neighbor (RNN) clustering for clustering terms. The reciprocal nearest neighbor clustering consists of two items, one is the nearest neighbor of the other one according to the similarity measure. Lewis chose a probabilistic approach to text categorization and his results were inferior to those obtained by word indexing.

Baker & McCallum [1] introduced the distributional clustering to document classification with a Naïve Bayes classifier. Differed from other similarity metrics, the distributional clustering calculated the probability distribution over the class introduced by the different words to be clustered. The Kullback-Leibler divergence, which is an information theoretic measurement, was used to exactly measure the difference between two probability distributions. Baker and McCallum found that the distributional clustering is better than feature selection with regard to preserving the information contained in redundant features. The experimental results showed that the categorization based on word clustering can maintain good performance and keep a significantly more compact representation.

III. Algorithm Description

A. Motivation

Baker and McCallum [1] considered the probability distribution of a particular word w_t over classes C can be described as $P(C|w_t)$. When word w_t and w_s are clustered together, the new distribution is the weighted average of the distribution of word w_t and w_s. Table I lists the term frequencies of three features in 10 categories and the class probability distributions for these features. The numbers in the parentheses are the class probability distributions. The class distribution curves of three features are shown in Fig 1. The horizontal axis is the list of class labels. The vertical axis indicates the probability of a term over each class. The curve in Fig 1 can be interpreted as the probability distribution of a feature against classes. It can be seen from Fig 1 that the shape of the probability distribution of feature 1 is quite similar to that of feature 2 and dissimilar to that of feature 3. Thus the feature 1 and feature 2 can be clustered together according to the idea of distributional clustering.

The probability distributions of words over each class are considered as the similarity metric in distributional clustering of words [1]. Inspired by the probability distributions of one word over each class, we believed the differences of term frequencies of a feature occurring in various categories could be regarded as the level of the contributions of the feature to each class in the context of document classification. The term frequencies of a feature occurring in various categories are the points in a two-dimensional space, which consists of the term frequency and class label. Similar to the class probability distribution curve in the distributional clustering, the term frequencies of a feature occurring in various categories are joined and form a contribution line in term frequency against class label plane. The contribution curves of three features listed in Table I are shown in Fig 2. The horizontal axis has ticks for list of class labels. The vertical axis

indicates the term frequencies of a feature occurring in various classes. It can be seen from Fig 2 that the differences among relative contributions of three features are equal to zero if their positions in two-dimensional space are not considered. In respect of the document classification, the relative contributions of three features to each class are identical. Therefore, these features should be clustered together.

TABLE I

THE TERM FREQUENCIES AND THE PROBABILITIES OF THREE FEATURES OVER 10 CATEGORIES.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Feature 1	115(0.2174)	103(0.1947)	104(0.1966)	200(0.3781)	190(0.3592)	109(0.2060)	114(0.2155)	103(0.1947)	107(0.2023)	101(0.1909)
Feature 2	85(0.2191)	73(0.1881)	74(0.1907)	170(0.4381)	160(0.4124)	79(0.2036)	85(0.2191)	74(0.1907)	75(0.1933)	73(0.1881)
Feature 3	15(0.0605)	3(0.0121)	4(0.0161)	100(0.4032)	90(0.3629)	9(0.0363)	15(0.0605)	4(0.0161)	5(0.0202)	3(0.0121)

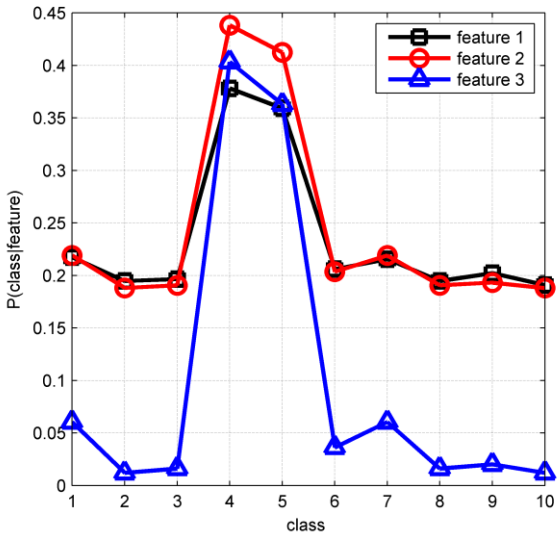

Fig 1. The probability distribution curves of feature 1, 2 and 3 over various categories.

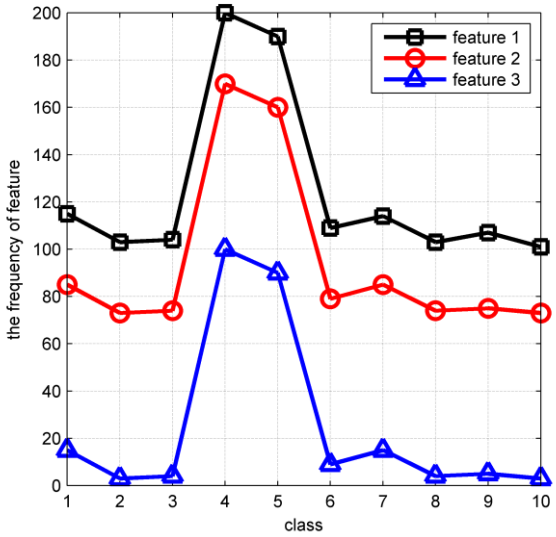

Fig 2. The contribution curve of feature 1, 2 and 3 for classification.

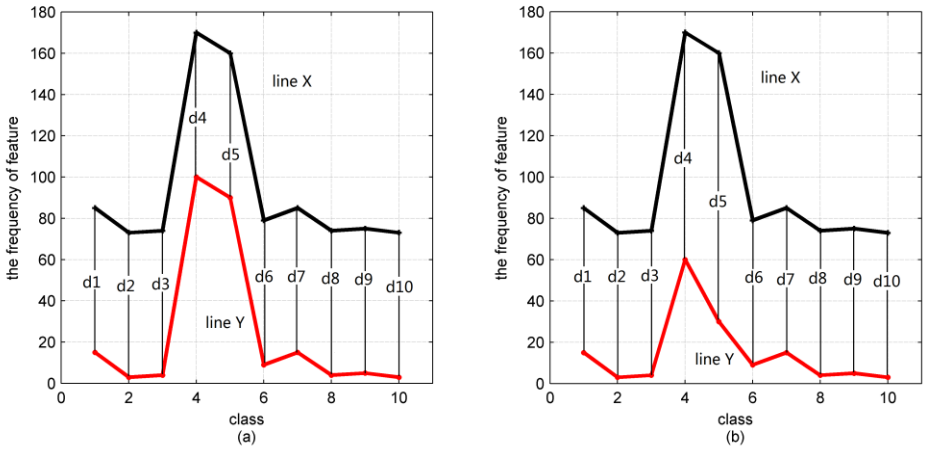

Fig 3. Two pairs of contribution curves, one is similar and the other is dissimilar.

B. Similarity measurement

It is assumed that the contribution curves of two features can be described as two vector $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$; x_i and y_i, $i \in [1, n]$, $n \geqslant 2$ are the term frequencies of two different features occurring in category c_i, respectively. We believe that if the distances ($d_i = x_i - y_i$) between corresponding elements of two vectors are equal or approximately equal, the relative contribution of the two features for classification should be similar to each other. Fig 3 shows two pairs of contribution curves of two features. d_i, $i \in [1, 10]$, is the distance between corresponding elements of two vectors. The two contribution curves drew in Fig 3 (a) are similar to each other because of that the distances between corresponding elements of two vectors are equal; the two curves drew in Fig 3 (b) are not similar to each other because of that d_4 and d_5 are significantly greater than others.

In this paper, we proposed two approaches to measure the similarity between contribution curves of two features.

In the first approach, we use the variance of the differences between corresponding elements in two vectors to measure the similarity between two contribution curves, called Relative Contribution 1. The smaller the variance is, the more similar the relative contributions of two features are. Specially, if the variance is equal to zero, the relative contribution of two features are identical. The measure of the similarity between two vector $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$ corresponding to two features is listed as follows:

$$d(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}\left[\left(x_i - y_i\right) - \frac{1}{n}\sum_{i=1}^{n}\left(x_i - y_i\right)\right]^2$$

In the second approach, we use the difference between two adjacent elements in one vector as the relative contribution of these elements to categorization, called Relative Contribution 2. We firstly calculate the difference between two adjacent elements in one vector, and then calculate the difference between two vectors. The measure of the similarity between two vector $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$ corresponding to two features is listed as follows:

$$d(X,Y) = \frac{1}{n}\sum_{i=1}^{n-1}\left[\left(x_i - x_{i+1}\right) - \left(y_i - y_{i+1}\right)\right]^2$$

C. Document representation

After all the terms are grouped into K clusters, every document in the training set and test set will be represented as a vector in which the words occurring in the document will be mapped into K clusters. For instance, Table II is the list of words and their term frequencies in one document. Table III indicates 10 clusters created by them. In our experiment, the document is represented as a vector, where the element value is the sum of term frequencies of terms included in the same cluster. Thus, the document listed in Table II can be represented by {6, 13, 6, 1, 7, 2, 1, 2, 3, 2}.

IV. Experimental Setup

A. Datasets and preprocessing

For the document data which is expected to be available for term clustering, it is first converted into a term list. In the experiment, the stop-words are removed and the stemmer program processes the different forms of the same word root [15]. We used a stop-words list, which contains 571 words. For the stemmer program, we used the Porter's stemming algorithm (http://tartarus.org/~martin/PorterStemmer/). The details are shown as follows:

1. Identify all words in the training set and make a list of terms.
2. Remove stop-words and apply the stemming algorithm to each word in the list.
3. Build a matrix M: each column (j) is a category in the training set, and each row (i) is a term in the list. The $M(i,j)$ is the sum of term frequency of the ith term occurs in the jth category.

In order to evaluate the performance of the proposed method, three benchmark datasets - 20-Newsgroups, Reuters-21578 and Industry Sector - were used in our study.

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 evenly assigned to 20 different UseNet groups. In our study, we only consider three categories such as "talk.politics.guns", "talk.politics.mideast" and "talk.politics.misc". We ignore the UseNet header and only consider the content of the document when tokenizing the document.

TABLE II

LIST OF WORDS AND THEIR TERM FREQUENCIES IN ONE DOCUMENT.

word	number	word	number	word	number
hong	2	share	3	own	1
kong	2	total	1	fund	1
firm	2	outstanding	1	publicly	1
stake	3	common	2	held	1
washington	1	stock	2	zealand	1
industrial	2	filing	1	company	1
equity	2	security	1	bought	2
pacific	1	exchange	1	disclose	1
investment	2	commission	1	earlier	1
raise	1	principally	1	month	1

TABLE III

LIST OF 10 CLUSTERS CREATED BY THE WORDS OCCURRING IN TABLE II. THE NUMBER IN THE PARENTHESES IS THE SUM OF TERM FREQUENCIES OF WORDS OCCURRING IN THE SAME CLUSTER.

C-1(6)	C-2(13)	C-3(6)	C-4(1)	C-5(7)	C-6(2)	C-7(1)	C-8(2)	C-9(3)	C-10(2)
hong	stake	company	outstanding	common	month	total	disclose	held	principally
kong	exchange	own		publicly	earlier		security	raise	commission
zealand	investment	firm		pacific				filing	
washington	fund	industrial		share					
	stock								
	bought								
	equity								

The Reuters-21578 corpus contains 21578 stories taken from the Reuters newswire. All stories are non-uniformly divided into 135 categories. In this paper, we only consider the top 10 categories such as “Earn”, “Acquisition”, “Money-fx”, “Grain”, “Crude”, “Trade”, “Interest”, “Wheat”, “Ship” and “Corn”. There are 9982 documents in top 10 categories.

The Industry Sector corpus, made available by Market Guide Inc. (www.marketguide.com), consists of company web pages classified in a hierarchy of industry sectors [16]. There are 6440 web pages attached to 12 root classes and then hierarchically partitioned into 71 classes. In this paper, we only consider 7 root classes, such as “capital.goods.sector”, “conglomerates.industry”, “consumer.non-cyclical.sector”, “energy.sector”, “healthcare.sector”, “transportation.sector” and “utilities.sector”.

B. Clustering algorithm

Clustering [17, 18], which is a method of unsupervised learning, is a common technique for statistical data analysis applied in many fields, such as machine learning, data mining, pattern recognition and information processing. There exist many different approaches for clustering, such as Hierarchical Clustering, Nearest Neighbor Clustering, K-means Clustering [19] and Expectation Maximization, and so on. The partitions generated by hierarchical clustering algorithm are more versatile than those generated by other clustering algorithms [20]. In cluster-based document retrieval, the hierarchical clustering algorithm performed better than other clustering algorithms [20]. The results of hierarchical clustering are usually presented in a dendrogram which represents the nested grouping of data and similarity levels at which groupings change [20]. In hierarchical clustering, the number of cluster need not be known in advance and can be determined based on the dendrogram by the really requirements [21]. The strategies of hierarchical clustering generally fall into two types. One is agglomerative hierarchical clustering that begins with each element as a separate cluster and merge them into

larger clusters. The other is divisive hierarchical clustering that begins with the whole set and proceed to divide it into smaller clusters. In this paper, the agglomerative hierarchical clustering algorithm developed by Hoon, et al. [22] was adopted.

C. Similarity Measure

The similarity measures, which are used to compare with the proposed method, are detailed in this section.

The KL divergence to mean, which is the average of the KL divergence of each distribution, is used in distributional clustering [1]. Baker and McCallum used the weighted average instead of the simple average. The weighted average KL divergence is defined as follows:

$$d = P(w_t)D(P(C \mid w_t) \,\|\, P(C \mid w_t \vee w_s)) + P(w_s)D(P(C \mid w_s) \,\|\, P(C \mid w_t \vee w_s))$$

where $P(w_t)$ and $P(w_s)$ are the probability of the word w_t and w_s occurring in the training set, respectively; $P(C|w_t)$ and $P(C|w_s)$ are the contribution of the word w_t and w_s to classification, respectively; $P(C|w_t \vee w_s)$ is the contribution of cluster in which the word w_t and w_s are combined to classification. $D(P(C|w_t)||P(C|w_t \vee w_s))$ is the measure of inefficiency that occurs when messages are sent according to one distribution, $P(C|w_t)$, but encoded with a code that is optimal for a different distribution, $P(C|w_t \vee w_s)$.

The City-block metric, which is known as the Manhattan distance, is the sum of distances along each element in vector $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$.

$$d = \frac{1}{n}\sum_{i=1}^{n} | x_i - y_i |$$

The Euclidean metric is one of the most common types of distance. It is the geometric distance in the multidimensional space. The Euclidean distance between two points in n-dimension space $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ can be computed as follows:

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

D. Classifiers

Many classifiers are used in text categorization in recent years, such as Naïve Bayes (NB), K-nearest neighbor (KNN), Support Vector Machines (SVM)[23], Rocchio, and so on. Compared with the state-of-art methods, Support Vector Machines is a higher efficient classifier in text categorization. There is much empirical support for using Support Vector Machines for text categorization [6, 24, 25]. The published results show that KNN is quite effective and its performance is comparable to that obtained by SVM [2]. In this paper, we use SVM and KNN to compare the performance of various clustering methods.

Support Vector Machines is based on the structural risk minimization principle for computational learning theory, and it was originally developed by Drucker, et al. [26] and applied to text categorization by Joachims [3]. Since Joachims [3] thought that most of the text categorization problems are linearly separable, the linear kernel for SVM is selected. In this study, we use LIBSVM toolkit [27]. C-SVM [28, 29] is selected and the penalty parameter C is 1.

KNN [30, 31] is a simple machine learning algorithm that makes decision depending on the major category labels attached to the k training documents which are similar to the test object, and it is a type of instance-based classifier or lazy learner, since the decision is made until all the objects in the training set are scanned [2]. We used k=30 in this experiment, and the cosine distance was used as the measure of document similarity.

E. Evaluations

The classification effectiveness in text categorization is usually measured in terms of the precision (P) and recall (R) [2] which are originally defined for binary classification [24]. To compute the averaged estimates in multiclass classification context, the micro-averaging method and macro-averaging method are used [25]. The micro-averaging measure and macro-averaging measure are computed as follows:

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

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

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

$$P_{macro} = \frac{\sum_{i=1}^{|C|} P_i}{|C|} \qquad R_{macro} = \frac{\sum_{i=1}^{|C|} R_i}{|C|}$$

$$F1_{macro} = \frac{2R_{macro}P_{macro}}{R_{macro} + P_{macro}}$$

where TP_i is the amount of the documents which are correctly classified to category c_i; FP_i is the amount of the documents which are misclassified to the category c_i; FN_i is the amount of the documents which are belong to category c_i and misclassified to other categories; $|C|$ is the amount of the categories.

The accuracy, which is defined to be the percentage of correctly labeled documents in test set, is widely used in text categorization [6, 15, 32-34]. The formula of the accuracy is defined as follows:

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

F. Validation

In order to validate the text representation method based on word clusters created by the proposed method, the 5×2 fold cross validation [35] is used in this paper. We perform 5 replications of 2-fold cross validation. In each replication, the documents in corpus are randomly partitioned into two equal-sized subsets (A and B). The learning algorithm is trained on subset A and tested on subset B, and then trained on subset B and tested on subset A. So 10 performance estimates are produced in 5×2 fold cross validation.

V. RESULTS

A. Results on 20-Newsgroups corpus

Table IV and Table V show the micro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering algorithm on 20-newsgroups. It can be seen from Table IV that the performance of the proposed methods outperforms that of the other similarity measures in terms of micro F1 when the number of clusters is 100, 200 or 300, respectively. Table V shows that the performance of the Relative Contribution 1 is superior to that of the other similarity measures when the number of the clusters is 800, 900 or 1000. The macro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering algorithm on 20-newsgroups are listed in Table VI and Table VII. It can be seen from Table VI that the performance of the proposed methods outperforms that of the other similarity measures in terms of macro F1 when the number of clusters is 100, 200 or 300, respectively. Table VII shows that the performance of the Relative Contribution 1 is superior to that of Cityblock and Euclidean except for the number of the clusters is 500 or 600. Moreover, the performance of the Relative Contribution 1 is superior to that of WeightedKLD when the number of the clusters is 800, 900 or 1000. Fig 4 indicates the accuracy curves of SVM and KNN when five similarity measures are used on 20-newsgroups, respectively. It can be seen from Fig 4(a) that the proposed methods achieve better perfomance with fewer clusters. Fig 4(b) indicates that the accuracy performance of the proposed methods is superior to that of other similarity measures when the number of clusters is greater than 700.

B. Results on Reuters-21578 corpus

Table VIII and Table IX show the micro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering

TABLE IV

THE MICRO F1 OF SUPPORT VECTOR MACHINES BASED ON 20-NEWSGROUPS.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	87.00	85.60	84.63	84.63	84.80	84.70	84.98	85.14	84.98	84.92
Relative Contribution 2	86.37	84.81	85.92	84.53	84.73	84.94	84.99	84.95	85.25	85.13
WeightedKLD	85.98	84.06	85.57	85.46	85.61	85.66	85.44	85.32	85.31	85.26
Cityblock	85.93	83.85	84.88	85.32	85.18	86.32	86.26	86.42	86.52	86.77
Euclidean	85.74	84.18	85.54	85.32	85.57	85.95	86.45	86.67	86.37	86.77

algorithm on Reuters-21578. Table X and Table XI show the macro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering algorithm on Reuters-21578. Fig 5 draws the accuracy curves of SVM and KNN when five similarity measures are used in hierarchical clustering on Reusters-21578, respectively. It can be seen from Table VIII - XI and Fig 5 that the performance of the proposed measures is only inferior to that of the WeightedKLD and superior to that of Cityblock and Euclidean.

TABLE V

THE MICRO F1 OF K-NEAREST NEIGHBOR BASED ON 20-NEWSGROUPS.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	80.31	82.73	81.91	80.99	80.87	80.95	82.52	84.36	84.1	83.87
Relative Contribution 2	80.67	81.8	81.14	80.67	81.01	81.29	80.83	81.52	83.03	82.73
WeightedKLD	84.10	83.93	84.12	83.34	83.70	83.02	82.77	82.48	82.87	83.29
Cityblock	77.50	78.82	81.28	79.98	80.51	82.15	82.15	83.04	82.87	82.34
Euclidean	79.26	79.23	80.96	79.97	81.36	82.34	82.35	83.24	83.42	82.90

TABLE VI

THE MACRO F1 OF SUPPORT VECTOR MACHINES BASED ON 20-NEWSGROUPS.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	86.98	85.58	84.59	84.60	84.77	84.68	84.92	85.09	84.89	84.85
Relative Contribution 2	86.34	84.79	85.89	84.47	84.66	84.91	84.94	84.91	85.21	85.09
WeightedKLD	85.92	83.99	85.47	85.37	85.54	85.62	85.37	85.23	85.23	85.20
Cityblock	85.90	83.82	84.85	85.29	85.15	86.29	86.21	86.36	86.48	86.73
Euclidean	85.70	84.12	85.51	85.29	85.53	85.92	86.41	86.62	86.33	86.72

TABLE VII

THE MACRO F1 OF K-NEAREST NEIGHBOR BASED ON 20-NEWSGROUPS.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	80.12	82.60	81.63	80.79	80.75	80.84	82.40	84.31	84.04	83.80
Relative Contribution 2	80.39	81.55	80.79	80.25	80.63	81.17	80.68	81.39	82.90	82.59
WeightedKLD	84.04	83.83	84.07	83.25	83.61	82.92	82.64	82.34	82.77	83.18
Cityblock	76.69	77.86	80.69	79.19	79.88	81.63	81.54	82.45	82.39	81.72
Euclidean	78.81	78.47	80.32	79.16	80.90	81.93	81.82	82.89	83.01	82.40

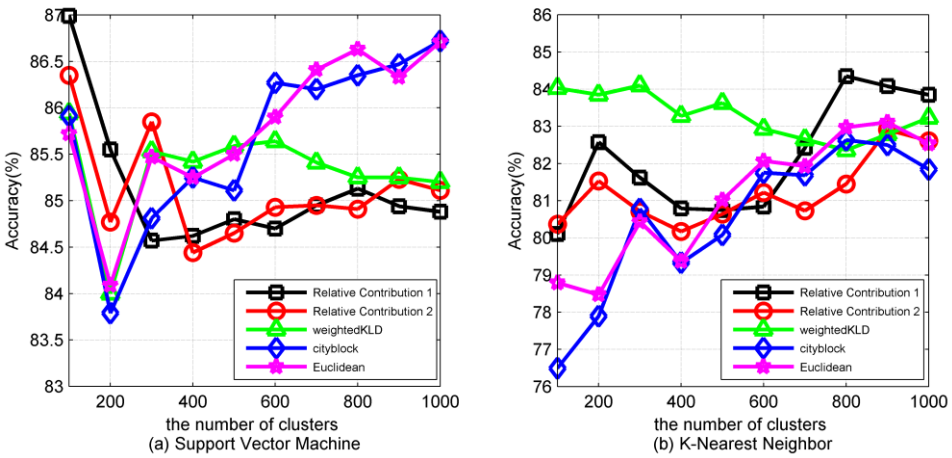

Fig 4. the accuracy curves of Support Vector Machines and K-Nearest Neighbor used on 20-Newsgroups, respectively.

TABLE VIII

THE MICRO F1 OF SUPPORT VECTOR MACHINES BASED ON REUTERS-21578.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	62.49	63.26	63.15	63.51	63.44	63.46	63.40	63.40	63.13	63.30
Relative Contribution 2	62.27	62.92	63.04	63.28	63.01	62.89	63.18	63.45	63.42	63.41
WeightedKLD	62.32	63.69	63.82	63.98	63.78	63.90	63.66	63.84	63.68	63.71
Cityblock	56.93	59.03	61.14	61.63	61.90	62.38	62.79	62.92	63.44	63.03
Euclidean	62.84	62.57	61.30	61.70	62.34	62.84	62.94	62.94	63.15	63.37

TABLE IX

THE MICRO F1 OF K-NEAREST NEIGHBOR BASED ON REUTERS-21578.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	58.79	60.14	59.07	58.61	58.44	58.37	58.07	58.12	57.86	57.42
Relative Contribution 2	58.35	61.21	59.66	60	59.53	59.41	59.38	59.02	58.65	57.48
WeightedKLD	59.93	63.75	63.74	63.27	62.88	62.27	61.76	61.08	61.08	60.80
Cityblock	40.06	42.92	44.56	46.17	47.63	49.33	50.49	51.13	52.50	53.16
Euclidean	57.29	43.31	45.99	47.81	49.02	50.95	52.28	53.00	53.94	55.11

TABLE X

THE MACRO F1 OF SUPPORT VECTOR MACHINES BASED ON REUTERS-21578.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	59.90	59.88	60.03	60.01	60.02	59.92	59.78	59.79	59.71	59.67
Relative Contribution 2	59.06	59.79	60.01	60.04	59.64	59.70	59.72	59.82	59.75	59.61
WeightedKLD	59.56	60.74	60.57	60.65	60.50	60.31	60.15	60.17	60.16	60.09
Cityblock	53.44	55.28	57.05	57.82	58.04	58.38	58.64	58.85	59.13	59.26
Euclidean	59.73	59.53	57.46	57.84	58.39	58.52	58.91	59.13	59.35	59.57

TABLE XI

THE MACRO F1 OF K-NEAREST NEIGHBOR BASED ON REUTERS-21578.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	55.62	57.00	56.04	55.74	56.01	55.70	55.31	55.41	54.97	54.77
Relative Contribution 2	55.21	58.24	56.42	56.86	56.33	56.47	56.49	56.18	55.70	54.83
WeightedKLD	57.89	61.19	61.38	60.94	60.67	59.88	59.32	58.76	58.50	58.19
Cityblock	37.48	40.59	41.93	43.64	45.25	47.17	48.41	49.07	50.50	51.23
Euclidean	54.47	41.05	43.62	45.71	46.93	49.14	50.37	51.16	52.23	53.58

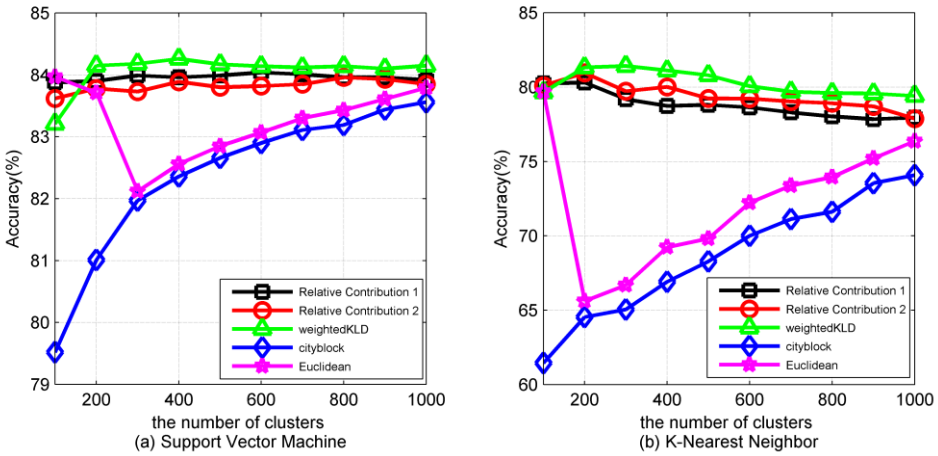

Fig 5. the accuracy curves of Support Vector Machines and K-Nearest Neighbor used on Reuters-21578, respectively.

C. Results on Industry-Sector corpus

Table XII and Table XIII show the micro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering algorithm on Industry Sector corpus. Table XIV and Table XV show the macro F1 measures of Support Vector Machines and K-Nearest Neighbor when five similarity measures are used in hierarchical clustering algorithm on Industry Sector corpus. It can be seen from Table XII and Table XIV that the performance of SVM combined with the Relative Contribution 2 is superior to that of other similarity measures when the number of clusters is 900 or 1000. Table XIII and Table XV indicate that the performance of KNN combined with the Relative Contribution 1 is superior to that of other similarity measures when the number of clusters is 800 or 900. Fig 6 shows the accuracy of SVM and KNN when five similarity measures are used in hierarchical clustering on Industry Sector corpus. Fig 6(a) indicates that the accuracy curve of SVM combined with the proposed method is higher than that with other methods when the number of clusters is 600, 700 or 800. Fig 6(b) shows that the accuracy curve of KNN combined with the Relative Contribution 1 is higher than that with the other methods when the number of clusters is 700, 800 or 900.

TABLE XII

THE MICRO F1 OF SUPPORT VECTOR MACHINES BASED ON INDUSTRY-SECTOR.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	48.72	50.78	51.51	53.27	53.41	55.44	56.67	58.59	57.85	59.06
Relative Contribution 2	49.73	51.11	51.74	54.82	56.24	55.71	57.06	58.13	59.19	60.19
WeightedKLD	51.81	57.51	56.17	56.13	56.96	56.90	57.75	58.83	59.18	59.26
Cityblock	46.48	50.11	52.18	53.48	54.19	54.22	54.64	54.82	55.80	57.15
Euclidean	49.47	50.25	51.38	51.81	54.64	55.06	57.69	57.70	59.04	59.42

TABLE XIII

THE MICRO F1 OF K-NEAREST NEIGHBOR BASED ON INDUSTRY-SECTOR.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	39.08	40.65	43.84	44.13	45.37	48.04	48.18	50.77	51.56	52.08
Relative Contribution 2	38.09	39.05	39.62	44.89	44.53	48.47	49.09	49.41	49.57	52.32
WeightedKLD	46.10	49.57	48.87	47.90	47.56	45.12	45.83	47.91	48.87	49.52
Cityblock	32.56	36.38	40.37	38.71	42.12	43.38	42.82	45.04	42.99	45.90
Euclidean	36.20	37.88	41.22	44.49	45.50	47.36	48.51	50.70	50.99	52.69

TABLE XIV

THE MACRO F1 OF SUPPORT VECTOR MACHINES BASED ON INDUSTRY-SECTOR.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	47.87	50.07	51.04	52.59	52.96	54.95	55.90	57.80	57.04	58.19
Relative Contribution 2	48.07	50.21	51.15	53.90	55.29	54.63	55.90	57.13	58.16	58.98
WeightedKLD	49.39	54.69	53.76	54.40	55.30	55.35	56.32	57.34	57.76	58.07
Cityblock	45.22	49.36	51.60	53.01	53.85	53.78	54.19	54.40	55.30	56.62
Euclidean	48.35	49.75	50.92	51.31	54.19	54.65	57.07	56.81	58.08	58.64

TABLE XV

THE MACRO F1 OF K-NEAREST NEIGHBOR BASED ON INDUSTRY-SECTOR.

	100	200	300	400	500	600	700	800	900	1000
Relative Contribution 1	35.42	36.70	38.54	40.09	41.21	42.78	43.55	45.62	46.37	47.18
Relative Contribution 2	34.12	35.68	35.07	39.53	40.31	42.53	43.23	43.67	44.28	47.38
WeightedKLD	43.39	46.77	47.11	45.44	45.12	42.60	43.68	45.05	45.54	46.18
Cityblock	28.81	33.17	34.94	35.61	38.21	39.34	39.25	40.03	39.08	41.31
Euclidean	32.40	34.29	36.71	38.16	41.01	42.11	43.08	44.94	45.30	47.38

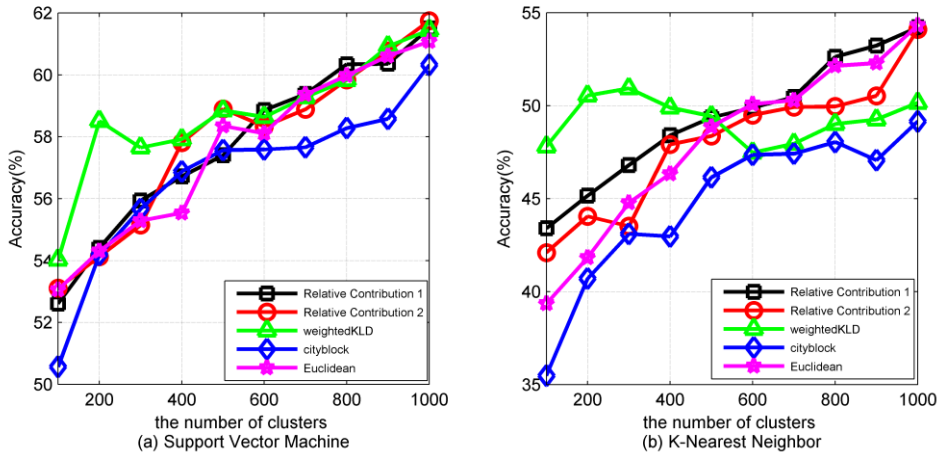

Fig 6. the accuracy curves of Support Vector Machines and K-Nearest Neighbor used on Industry-Sector, respectively.

VI. Analysis and Discussion

The evaluation for clustering results to find the partitioning that best fits the underlying data is one of the most important issues in cluster analysis [36, 37]. There are many cluster validation indices that have been proposed to validate the quality of the clusters in the literature [38]. Davies and Bouldin [39] presented a validation index which can infer the appropriateness of various divisions of the data. The Davies-Bouldin index has a low computational complexity. So we chose it to validate the quality of the clusters produced by the proposed method. Its formula is listed as follows:

$$DB = \frac{1}{M}\sum_{i=1}^{M}\max_{j=1,\ldots,M;\, j\neq i}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right)$$

where M is the number of the clusters; σ_i (σ_j) is the average distance of all elements in cluster i (j) to their cluster center c_i (c_j); $d(c_i, c_j)$ is the distance between the cluster centers c_i and c_j. The range of DB is [0, ∞] and the lower value is for a good clustering [38].

Table XVI lists the Davies-Bouldin indices of the clusters when various similarity measures are used and all words in feature vector space are clustered into k clusters. The k is equal to 100, 200, 300, 400, 500, 600, 700, 800, 900 or 1000. Due to the value of that is too large to display, the Davies-Bouldin indices of clusters generated by City-block and Euclidean are not listed in Table XVI. It can be seen from Table XVI that the quality of clusters generated by the proposed method is the best. However, there exists a problem that the quality of term cluster is not consistent with the performance of classifiers based on the term cluster.

We ran our experiments on an Intel Core2 Q6600 2G RAM PC under windows XP. Due to the limitation of memory, the size of the feature space must be less than 25000; otherwise, the clustering software will prompt that the memory is insufficient for clustering. In our experiments, we chose a part of corpora (20-newsgroups, reuters-21578 and industry sector) to ensure that the size of the feature space is less than 20000. The clustering based on the proposed method run about one hour.

TABLE XVI

THE DAVIES-BOULDIN INDICES OF THE CLUSTERS GENERATED BY VARIOUS SIMILARITY MEASURES ON THREE TEXT CORPORA.

Dataset	Similarity measure	100	200	300	400	500	600	700	800	900	1000
20-newsgroups	Relative Contribution 1	0.1792	0.1946	0.1752	0.1649	0.1606	0.1517	0.1666	0.1607	0.1477	0.1192
	weightedKLD	0.2550	0.2106	0.2110	0.1897	0.1775	0.1719	0.1610	0.1487	0.1353	0.1218
Reuters-21578	Relative Contribution 1	0.2633	0.2217	0.2028	0.1930	0.1941	0.1814	0.1809	0.1766	0.1823	0.1787
	weightedKLD	0.2643	0.2313	0.2498	0.2621	0.2632	0.2638	0.2655	0.2609	0.2661	0.2543
Industry Sector	Relative Contribution 1	0.2930	0.2560	0.2272	0.2288	0.2077	0.1978	0.1879	0.1888	0.1796	0.1669
	weightedKLD	0.3104	0.3103	0.3297	0.3465	0.3219	0.3293	0.3097	0.2890	0.2782	0.2684

VII. Conclusion

We proposed two new similarity measure methods, which use the relative contribution of a feature for categories as the measure criterion. The proposed similarity metric is used in term clustering for text categorization and can reduce the feature space by one to three orders of magnitude while losing only a few percent in classification performance.

We evaluated the proposed methods on three benchmark corpora (20-newgroups, reuters-21578 and industry sector), using two classification algorithms (Support Vector Machines, K-Nearest Neighbor), and compared with three well-known similarity measures (weighted average KL divergence, City-block, Euclidean). The results show that the performance of the proposed methods is comparative with that of other methods when Support Vector Machines is used; the proposed methods significantly outperform Euclidean and City-block，and achieve comparative performance with weighted average KL divergence when K-Nearest Neighbor classifier is used. Moreover, the quality of the clusters generated by the proposed method is the best. In the future, the relationship between the quality of a cluster and the performance of classifiers based on the cluster is a research focus for us.

References

[1] L. D. Baker and A. K. McCallum, "Distributional clustering of words for text classification," in proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, Melbourne, Australia, 1998.

[2] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, pp. 1-47, Mar. 2002.

[3] T. Joachims, "Text categorization with Support Vector Machines: Learning with many relevant features," in *Machine Learning: ECML-98*. vol. 1398, C. Nédellec and C. Rouveirol, Eds., ed: Springer Berlin / Heidelberg, 1998, pp. 137-142.

[4] C. Lee and G. G. Lee, "Information gain and divergence-based feature selection for machine learning-based text categorization," *Information Processing & Management*, vol. 42, pp. 155-165, 2006.

[5] D. Fragoudis, *et al.*, "Best terms: an efficient feature-selection algorithm for text categorization," *Knowledge and Information Systems*, vol. 8, pp. 16-33, 2005.

[6] R. Bekkerman, *et al.*, "Distributional word clusters vs. words for text categorization," *J. Mach. Learn. Res.*, vol. 3, pp. 1183-1208, 2003.

[7] A.-M. Hisham, "A New Text Categorization Technique Using Distributional Clustering and Learning Logic," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, pp. 1156-1165, 2006.

[8] N. Slonim and N. Tishby, "The power of word clusters for text classification," in *proceedings of ECIR-01, 23rd European Colloquium on Information Retrieval Research*, Darmstadt, Germany, 2001.

[9] M. McGill, "An Evaluation of Factors Affecting Document Ranking by Information Retrieval Systems," Technical report, Syracuse University School of Information Studies, 1979.

[10] S. Ramaswamy and K. Rose, "Adaptive Cluster Distance Bounding for High-Dimensional Indexing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, pp. 815-830, 2011.

[11] P.-E. DANIELSSON, "Euclidean Distance Mapping," *COMPUTER GRAPHICS AND IMAGE PROCESSING*, vol. 14, pp. 227-248, 1980.

[12] R. M. C. R. de Souza and F. d. A. T. de Carvalho, "Clustering of interval data based on city-block distances," *Pattern Recognition Letters*, vol. 25, pp. 353-365, 2004.

[13] J. D'Hondt, *et al.*, "Pairwise-adaptive dissimilarity measure for document clustering," *Information Sciences*, vol. 180, pp. 2341-2358, 2010.

[14] D. D. Lewis, "An evaluation of phrasal and clustered representations on a text categorization task," in proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, Copenhagen, Denmark, 1992.

[15] E. Youn and M. K. Jeong, "Class dependent feature scaling method using naive Bayes classifier for text datamining," *Pattern Recognition Letters*, vol. 30, pp. 477-485, 2009.

[16] A. McCallum, *et al.*, "Improving Text Classification by Shrinkage in a Hierarchy of Classes," in *ICML-98*, 1998.

[17] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, pp. 651-666, 2010.

[18] Y. Zhao and G. Karypis, "Data clustering in life sciences," *Molecular Biotechnology*, vol. 31, pp. 55-80, 2005.

[19] J. Y. B. Tan, *et al.*, "Time Series Prediction using Backpropagation Network Optimized by Hybrid K-means-Greedy Algorithm," *Engineering Letters*, vol. 20, pp. 203-210, 2012.

[20] A. K. Jain, *et al.*, "Data clustering: a review," *ACM Comput. Surv.*, vol. 31, pp. 264-323, 1999.

[21] Y. Miin-Shen and W. Kuo-Lung, "A similarity-based robust clustering method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, , vol. 26, pp. 434-448, 2004.

[22] M. J. L. d. Hoon, *et al.*, "Open source clustering software," *Bioinformatics*, vol. 20, pp. 1453-1454 2004.

[23] I. Nurtanio, *et al.*, "Classifying Cyst and Tumor Lesion Using Support Vector Machine Based on Dental Panoramic Images Texture Features," *IAENG International Journal of Computer Science*, vol. 40, pp. 29-37, 2013.

[24] T. Joachims, *Learning to Classify Text Using Support Vector*

Machines: Methods, Theory and Algorithms: Kluwer Academic Publishers 2002.

[25] H. Ogura, *et al.*, "Feature selection with a measure of deviations from Poisson in text categorization," *Expert Systems with Applications*, vol. 36, pp. 6826-6832, 2009.

[26] H. Drucker, *et al.*, "Support Vector Machines for Spam Categorization," *IEEE TRANSACTIONS ON NEURAL NETWORKS*, vol. 10, pp. 1048-1054, 1999.

[27] C.-C. Chang and C.-J. Lin. (2001). *LIBSVM : a library for support vector machines http://www.csie.ntu.edu.tw/~cjlin/libsvm*. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[28] C.-C. Chang and C.-J. Lin, "Training v-Support Vector Classifiers: Theory and Algorithms," *Neural Computation*, vol. 13, pp. 2119-2147, 2001.

[29] M. A. Davenport, *et al.*, "Controlling False Alarms with Support Vector Machines," presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2006.

[30] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, pp. 21-27, 1967.

[31] E. Uchino, *et al.*, "IVUS-Based Coronary Plaque Tissue Characterization Using Weighted Multiple k-Nearest Neighbor," *Engineering Letters*, vol. 20, pp. 211-216, 2012.

[32] Y. H. Li and A. K. Jain, "Classification of Text Documents," *The Computer Journal*, vol. 41, pp. 537-546, January 1, 1998 1998.

[33] M. Benkhalifa, *et al.*, "Integrating External Knowledge to Supplement Training Data in Semi-Supervised Learning for Text Categorization," *Information Retrieval*, vol. 4, pp. 91-113, 2001.

[34] A. Markov, *et al.*, "The hybrid representation model for web document classification," *International Journal of Intelligent Systems*, vol. 23, pp. 654-679, 2008.

[35] T. G. Dietterich, "Approximate Sratistical Tests for Comparing Supervised Classification Learning Algorithms," *Neural Computation*, vol. 10, pp. 1895-1923, 1998.

[36] M. Halkidi, *et al.*, "On Clustering Validation Techniques," *Journal of Intelligent Information Systems*, vol. 17, pp. 107-145, 2001.

[37] H. Alizadeh, *et al.*, "A Framework for Cluster Ensemble Based on a Max Metric as Cluster Evaluator," *IAENG International Journal of Computer Science*, vol. 39, pp. 10-19, 2012.

[38] S. Günter and H. Bunke, "Validation indices for graph clustering," *Pattern Recognition Letters*, vol. 24, pp. 1107-1113, 2003.

[39] D. L. Davies and D. W. Bouldin, "A Cluster Separation Measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 224-227, 1979.

Jie-Ming Yang received his MSc and PhD degree in computer applied technology from Northeast DianLi University, China in 2008 and Computer Science and Technology from Jilin University, China in 2013, respectively. His research interests are in machine learning and data mining.

Zhi-Ying Liu received her MSc degree in computer applied technology from Northeast DianLi University, China in 2005. His research interests are in information retrieval and personalized recommendation.

Zhao-Yang Qu received his MSc and PhD degree in computer science from Dalian University of Technology, China in 1988 and North China Electric Power University, China in 2012, respectively. His research interests are in artificial intelligence, machine learning and data mining.