Distributional Features for Text Categorization

Xiao-Bing Xue and Zhi-Hua Zhou, *Senior Member, IEEE*

***Abstract*— Text categorization is the task of assigning predefined categories to natural language text. With the widely used 'bag of words' representation, previous researches usually assign a word with values such that whether this word appears in the document concerned or how frequently this word appears. Although these values are useful for text categorization, they have not fully expressed the abundant information contained in the document. This paper explores the effect of other types of values, which express the distribution of a word in the document. These novel values assigned to a word are called *distributional features*, which include the compactness of the appearances of the word and the position of the first appearance of the word. The proposed distributional features are exploited by a *tfidf* style equation and different features are combined using ensemble learning techniques. Experiments show that the distributional features are useful for text categorization. In contrast to using the traditional term frequency values solely, including the distributional features requires only a little additional cost, while the categorization performance can be significantly improved. Further analysis shows that the distributional features are especially useful when documents are long and the writing style is casual.**

***Index Terms*— Text categorization, text mining, machine learning, distributional feature,** *tfidf*

I. Introduction

In the last ten years, content-based document management tasks have gained a prominent status in the information system field, due to the increased availability of documents in digital form and the ensuring need to access them in flexible ways [30]. Among such tasks, *Text Categorization* assigns predefined categories to natural language text according to its content. Text categorization has attracted more and more attention from researchers due to its wide applicability. Since this task can be naturally modeled as a supervised learning problem, many classifiers widely used in Machine Learning (ML) community have been applied, such as Naïve Bayes, Decision Tree, Neural Network, k Nearest Neighbor (kNN), Support Vector Machine (SVM) and AdaBoost. Recently, some excellent results have been obtained by SVM [13] and AdaBoost [28].

While a wide range of classifiers have been used, virtually all of them were based on the same text representation, 'bag of words', where a document is represented as a set of words appearing in this document. Values assigned to each word usually express whether the word appears in a document or how frequently this word appears. These values are indeed useful for text categorization. However, are these values enough?

Considering the following example, 'Here you are' and 'You are here' are two sentences corresponding to the same vector using the frequency related values, but their meanings are totally different. Although this is a somewhat extreme example, it clearly illustrates that besides the appearance and the frequency of appearances of a word, the distribution of a word is also important. Therefore, this paper attempts to design some *distributional features* to measure the characteristics of a word's distribution in a document. Note that the word 'feature' in 'distributional features' indicates the value assigned to a word, which is somewhat different from its usual meaning, i.e. the element used to characterize a document.

The first consideration is the compactness of the appearances of a word. Here, the *compactness* measures whether the appearances of a word concentrate in a specific part of a document or spread over the whole document. In the former situation, the word is considered as compact, while in the latter situation, the word is considered as less compact. This consideration is motivated by the following facts. A document usually contains several parts. If the appearances of a word are less compact, the word is more likely to appear in different parts and more likely to be related to the theme of the document. For example, consider Document *A* (NEWID=2367) and Document *B* (NEWID=7154) in Reuters-21578. Document *A* talks about the debate on whether expanding the 0/92 program or just limiting this program on wheat. Obviously, this document belongs to the category 'wheat'. Document *B* talks about the U.S. Agriculture Department's proposal on tighter federal standards about insect infections in grain shipments and this document belongs to the category 'grain' but not to the category 'wheat'. Let's consider the importance of the word 'wheat' in both documents. Since the content of Document *A* is more closely related to wheat than Document *B*, the importance of the word 'wheat' should be higher in Document *A* than in Document *B*. However, the frequency of this word is almost the same in both documents[1]. Therefore, the frequency is not enough to distinguish this difference of importance. Here, the compactness of the appearances of a word could provide a different view. In Document *A*, since the document mostly discusses the 0/92 program on wheat, the word 'wheat' appears in different parts of this document. In Document *B*, since the document mainly discusses the contents of the new standard on grain shipment and just one part of the new standard refers to wheat, the word 'wheat' only appears in one paragraph of this document. Thus, the compactness of the appearances of the word 'wheat' is lower in Document *A* than in Document *B*, which well expresses the importance of this word.

The second consideration is the position of the first appearance of a word. This consideration is based on an intuition

Manuscript received April, 2007.

The authors are with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China (e-mail: {xuexb, zhouzh}@lamda.nju.edu.cn).

[1]The normalized frequency is used here, which divides the number of the appearances of a word by the total number of words of a document.

that the author naturally mentions the important contents in the earlier parts of a document. Therefore, if a word firstly appears in the earlier parts of a document, this word is more likely to be important. Let's consider Document *A* (NEWID=3981) and Document *B* (NEWID=4679) in Reuters-21578. Document *A* belongs to the category 'grain' and talks about the heavy rain in Argentine grain area. Document *B* belongs to the category 'cotton' and discusses that China is trying to increase cotton output. Obviously, the word 'grain' should be more important in Document *A* than in Document *B*. Unfortunately, the frequency of the word 'grain' is even lower in Document *A* than in Document *B*. Now, let's consider the position of the first appearance of the word 'grain'. In Document *A*, it firstly appears in the title. It's not strange, since this document mainly talks about Argentine grain area. In Document *B*, the word 'grain' firstly appears at the end of the document. It's not strange either. Since the theme of this document is about increasing cotton output, the suggestion that the production of cotton be coordinated with other crops such as grain is indirectly related to this theme, so the author naturally mentioned this suggestion at the end of the document. Obviously, the position of the first appearance of a word could express the importance of this word to some extent.

Above all, when the frequency of a word expresses the intuition that *the more frequently a word appears, the more important this word is*, the compactness of the appearances of a word shows that *the less compactly a word appears, the more important this word is* and the position of the first appearance of a word shows that *the earlier a word is mentioned, the more important this word is*.

The contribution of this paper lies in:

- Design the distributional features for text categorization. Using these features can help improve the performance, while requiring only a little additional cost.
- How to use the distributional features is answered. Combining traditional term frequency with the distributional features results in improved performance.
- The factors affecting the performance of the distributional features are discussed. The benefit of the distributional features is closely related to the length of documents in a corpus and the writing style of documents.

The rest of this paper is organized as follows. Section 2 introduces some related work. Section 3 talks about how to extract the distributional features. Section 4 discusses how to utilize the distributional features. Section 5 reports experimental results. Finally, section 6 concludes this paper.

II. Related Work

When the features for text categorization are mentioned, the word 'feature' usually have two different but closely related meanings. One refers to which unit is used to represent a document or to index a document, while the other focuses on how to assign an appropriate weight to a given feature. Consider 'bag of words' as an example. Using the former meaning, the feature is a single word, while *tfidf* weighting is the feature given the latter meaning. This section will focus on previous researches about the features used for text categorization based on these two meanings. Other topics about text categorization can be found in a review paper [30].

For the first meaning, besides the single word, syntactic phrases have been explored by many researchers [9], [12], [20], [29]. A syntactic phrase is extracted according to language grammars. In general, experiments showed that syntactic phrases was not able to improve the performance of standard 'bag of word' indexing. Statistical phrases have also attracted much attention from researchers [5], [11], [22]. A statistical phrase is composed of s sequence of words that occur contiguously in text in a statistically interesting way [5], which is usually called n-*gram*. Here, n is the number of words in the sequence. When statistical phrases were used to enrich the text representation of the single word, better performance has been reported with the help of feature selection mechanism. Researchers also indicated that the short statistical phrase was more helpful than the long one [22]. In addition to phrases, other linguistic features such as POS-tag, word-senses and the synonym and hypernym relations in WordNet [10] were tried by researchers [23], [29]. Unfortunately, the improvement of performance brought by these linguistic features was somewhat disappointing. Word cluster was another promising feature for the first meaning [1], [2], [20]. A word's distribution on different categories was used to characterize a word [1], [2]. The clustering methods used by researchers included the agglomerative approach [1] and the recently proposed Information Bottleneck [2]. Experiments showed that the word cluster based representation outperformed the single word based representation sometimes.

Recently, Sauban and Phahringer [27] proposed a new text representation method, which explicitly exploited the information of word sequence. In their work, a discriminative score for every word was firstly calculated. Then, with every word inputted in sequence, a document was shown as a curve depicting the change of the accumulated scores. This curve was called 'Document Profiling'. Two different methods were used to turn a profile into a constant number of features. One was to sample from the profile with a fixed gap, while the other was to get some high-level summary information from the profile. Comparable results with the 'bag of words' representation were achieved with lower computational cost.

For the second meaning, the weight assigned to a given feature comes from two sources: intra-document and inter-document. The intra-document based weight uses information within a document, while the inter-document based weight uses information in the corpus. For *tfidf*, the tf part can be regraded as a weight from intra-document source, while the idf part is a weight from inter-document source.

There were relatively few researches on the intra-document based weight. Several variants of tf, such as the logarithmic frequency and the inverse frequency were used by researchers [16], [18]. The logarithmic frequency reflected that the intuition that the importance of a word should increase logarithmically instead of linearly with the increase of its frequency. The inverse frequency was derived in order to distribute term frequencies evenly on the interval from 0 to 1 [18]. Ko et al [15] used the importance of each sentence to weight the term frequency. Specifically, the importance of a sentence

was measured by two methods. One was to calculate the similarity between title and a given sentence, while the other one summed the importance of all words appearing in this sentence as the final importance. Given the importance of a sentence, for a word, a weighted term frequency was used to replace the original tf, where each appearance was weighted by the importance of the sentence where this appearance occurred.

For the inter-document based weight, researchers tried to improve the idf from both the unsupervised view and the supervised view. Researches from the unsupervised view didn't use the category information in training set. Leopold and Kindermann [18] proposed the Redundancy to measure the importance of a word, which quantifies the skewness of the distribution of this word's frequency in different documents. Lan et al [16] used the term relevance weight in their comparative study. The term relevance weight used the number of documents containing a word to divide the number of documents without this word, instead of the total number of documents in idf. Contrastingly, many researchers believed that the idf derived directly from text retrieval was not well suited for text categorization where the categories of training documents were available. In order to focus on the categorization task on hand, a lot of supervised weights were proposed. Shankar and Karypis [31] used a measure similar to Gini Index to calculate the discriminating power of each word. Debole and Sebastiani [7] modified the idf using some feature scoring functions widely used for feature selection such as Chi-square, Information Gain and Gain Ratio. The best finding was Gain Ratio, a variant of Information Gain. Soucy and Mineau [32] used a weighting method based on statistical confidence intervals. This method had an advantage of performing feature selection implicitly. In their work, a significant improvement over standard *tfidf* method was reported on benchmarks.

After talking about the related work in this area, a relatively accurate position can be found for our proposed distributional features. These features, which are the compactness of the appearances of a word and the position of the first appearance of a word, could be considered as a new weighting method using information within a document.

III. How to extract distributional features

Recall that the definitions of the two proposed distributional features are both based on the analysis of a word's distribution, thus modeling a word's distribution becomes the prerequisite for extracting required features.

A. Modelling a word's distribution

In this paper, a word's distribution is modeled by two steps: first, a document is divided into several parts; then, the distribution of a word is modeled as an array where each element records the number of appearances of this word in the corresponding part. The length of this array is the total number of the parts.

For the above model, how to define a part becomes a basic problem. According to Callan [4], there are three types of passages used in information retrieval[2]. Kim and Kim [14] discussed the advantages and disadvantages of these three types of passages. *Discourse Passage* is based on logic components of documents such as sentences and paragraphs. Discourse passage is intuitive, but it has two problems: the length of passages is inconsistent and sometimes no passage decoration is provided for documents. *Semantic Passage* is partitioned according to contents. This type of passage is more accurate, since each passage corresponds to a topic or subtopic, but its performance is heavily influenced by the effect of the partition algorithm. *Window Passage* is simply a sequence of words. Window passage is simple to implement, but it may break a sentence and the length of window is hard to choose. Considering efficiency, semantic passage is not used in the following experiments. Discourse passage and window passages with different sizes are explored respectively. Note that the window passage used in this paper is non-overlapped.

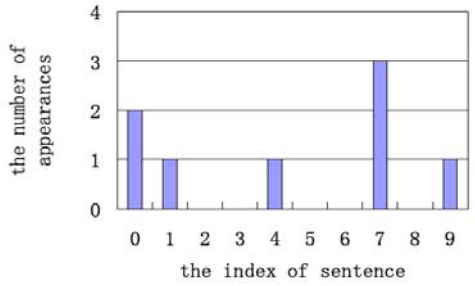


Fig. 1. The distribution of 'corn'.

Now, an example is given. For a document d with 10 sentences, the distribution of the word 'corn' is depicted as Fig. 1, then the distributional array for 'corn' is [2,1,0,0,1,0,0,3,0,1].

B. Extracting distributional features

Given a word's distribution, this subsection concentrated on implementing the two intuitively proposed distributional features.

For the compactness of the appearances of a word, three implementations are shown as follows. Note that, under the word distribution model mentioned above, the position of a word's appearance is just the index of the corresponding part.

- $ComPact_{PartNum}$ The number of parts where a word appears can be used to measure the concept of compactness. This is a natural implementation of the idea proposed in the introduction part. As what is mentioned, a word is less compact if it appears in different parts of a document.
- $ComPact_{FLDist}$ The distance between a word's first and last appearance is used to measure the compactness. It is motivated by the fact that, for a less compact word, the distance between the first mention and the last mention should be long. A slightly extreme example is the word that the author first mentions at the beginning of the

[2]Here, the meaning of 'passage' is the same as 'part' which is defined as any sequence of text from a document.

document and then mentions again at the end of the document.

- $ComPact_{PosVar}$ The variance of the positions of all appearances is used to measure the compactness. It is a natural implementation of the idea of compactness using the language of statistics. The mean position of all appearances is first calculated and then the mean distance between the position of each appearance and the mean position is calculated as the position variance.

For the position of the first appearance, this feature can be extracted directly from the proposed word distribution model.

Suppose in a document d containing n sentences, the distributional array of the word t is $array(t,d) = [c_0, c_1, ..., c_{n-1}]$. Then, the compactness ($ComPact$) of the appearances of the word t and the position of the first appearance ($FirstApp$) of the word t are defined, respectively, as follows:

$$FirstApp(t,d) = \min_{i \in \{0..n-1\}} c_i > 0?i : n \tag{1}$$

$$ComPact_{PartNum}(t,d) = \sum_{i=0}^{n-1} c_i > 0?1 : 0 \tag{2}$$

$$LastApp(t,d) = \max_{i \in \{0..n-1\}} c_i > 0?i : -1$$

$$ComPact_{FLDist}(t,d) = LastApp(t,d) - FirstApp(t,d) \tag{3}$$

$$count(t,d) = \sum_{i=0}^{n-1} c_i \qquad centroid(t,d) = \frac{\sum_{i=0}^{n-1} c_i \times i}{count(t,d)}$$

$$ComPact_{PosVar}(t,d) = \frac{\sum_{i=0}^{n-1} c_i \times |i - centroid(t,d)|}{count(t,d)} \tag{4}$$

Here, $exp = a?b : c$ means if the condition a is satisfied, the value of expression exp is b, otherwise the value is c.

The example in Fig. 1 is used again to illustrate how to calculate the distributional features.

$$\begin{aligned}
& FirstApp(corn,d) \\
= \; & min\{0,1,10,10,4,10,10,7,10,9\} = 0 \\
& ComPact_{PartNum}(corn,d) \\
= \; & 1+1+0+0+1+0+0+1+0+1 = 5 \\
& LastApp(corn,d) \\
= \; & max\{0,1,-1,-1,4,-1,-1,7,-1,9\} = 9 \\
& ComPact_{FLDist}(corn,d) \\
= \; & 9-0 = 9 \\
& count(corn,d) \\
= \; & 2+1+1+3+1 = 8 \\
& centroid(corn,d) \\
= \; & (2\times 0 + 1\times 1 + 1\times 4 + 3\times 7 + 1\times 9)/8 = 4.375 \\
& ComPact_{PosVar}(corn,d) \\
= \; & (2\times 4.375 + 1\times 3.375 + 1\times 0.375 + 3\times 2.625 \\
& +1\times 4.625)/8 = 3.125
\end{aligned}$$

Then, let's analyze the cost of extracting the term frequency and the distributional features. Suppose the size of the longest document in corpus is l, the size of the vocabulary is m, the biggest number of parts that a document contains is n and the number of documents in corpus is s. Usually, a memory block with size l is required for loading a document and an $m \times 1$ array is required for recording the number of appearances of each word in the vocabulary. When the scan of a document is completed, the term frequency can be directly obtained from the above array. In order to extract the distributional features, an additional $m \times n$ array is needed, since for each word, an $n \times 1$ array is used to record the distribution of this word. When the scan of a document is completed, Eqs. 1 to 4 are used to calculate the distributional features. No other additional cost is needed compared with extracting the term frequency. Overall, the additional computational cost for extracting the distributional features is $s \times m \times$ (Cost of Eqs. 1 to 4) and the additional storage cost is $m \times n$. It is worth noting that the above additional computational cost is the worst case, since practically the calculation is only required for words that appear at least once in a document. Actually, the number of such words in a document is significantly smaller than m. Generally, the additional computational and storage cost for extracting the distributional features is not big. The process of extracting the term frequency and the distributional features is illustrated in Fig. 2.

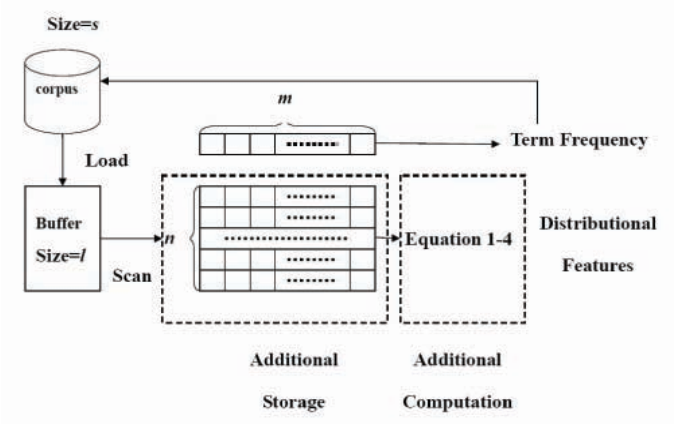


Fig. 2. The process of extracting term frequency and distributional features

The extraction of the distributional features can be efficiently implemented using the inverted index constructed for the corpus. Many retrieval systems such as Lemur and Indri[3] can support storing the positions of a word in a document in the index. Using such type of index, for a given word-document pair, we can not only obtain the frequencies of the word, but also the positions where the word appears. With the position information and the length of the document, it is easy to construct the distribution of this word and then the distributional features can be computed.

IV. How to utilize distributional features

Term Frequency in *tfidf* can be regarded as a value that measures the importance of a word in a document. As discussed in Introduction, the importance of a word can be measured not

[3]http://www.lemurproject.org/

only by its term frequency, but also by the compactness of its appearances and the position of its first appearance. Therefore, the standard *tfidf* equation can be generalized as follows:

$$tfidf(t,d) = Importance(t,d) \times idf(t) \tag{5}$$

Note that in Eq. 5 the standard idf term is used in order to focus on exploring the effect of the proposed distributional features. The idf term specifically designed for the distributional features will be explored in the future work.

When different features are involved, $Importance(t,d)$ corresponds to different values. When the feature is the frequency of a word, TermFrequency (TF) is used. When the feature is the compactness of the appearances of a word, ComPactness (CP) is used. When the feature is the position of the first appearance of a word, FirstAppearance (FA) is used. TF, CP and FA are calculated as follows:

$$TF(t,d) = \frac{count(t,d)}{size(d)} \tag{6}$$

$$CP_{PN}(t,d) = \frac{ComPact_{ParNum}(t,d)}{len(d)} \tag{7}$$

$$CP_{FLD}(t,d) = \frac{ComPact_{FLDist}(t,d)+1}{len(d)} \tag{8}$$

$$CP_{PV}(t,d) = \frac{ComPact_{PosVar}(t,d)+1}{len(d)} \tag{9}$$

$$FA(t,d) = f(FirstApp(t,d), len(d)) \tag{10}$$

$size(d)$ is the total number of words of Document d. $len(d)$ is the total number of parts of Document d. In Eqs 8 and 9, the numerator is added by 1 in order to ensure $CP(t,d) > 0$.

In Eq. 10, f is a weighting function used to assign different weights according to positions. Similar to Kim and Kim [14], four weighting functions are used in this paper as shown in Table I. Thus, four FA features are generated: FA_{GI}, FA_{GLI}, FA_{LL} and FA_{LVL}.

In Table I, p is the position of the part. These four weighting functions can be divided into two groups, global and local, as indicated by their names. Global functions used the absolute position, while local functions used the normalized position. The first three functions assume the importance decreases with the increase of position, while the last function LocalVLinear assumes the beginning and the end of a document have more importance than the body. Fig 3 shows the trends of these four functions in a document with 10 parts. Note that in this figure for each function the weight is normalized by its maximum weight to facilitate comparison. From this graph, it is clear that LocalVLinear is given such name due to its 'V'-like shape.

Finally, if a word t doesn't appear in Document d, $Importance(t,d)$ is set to 0, no matter what feature is used.

Since TF, CP and FA measure the importance of a word from different views, the combination of them may improve the performance. Ensemble learning technique [8] is exploited here. Specifically, a group of classifiers are trained based on different features. The label of a new document is decided by the combination of the outputs of these classifiers. Note that the outputs of each classifier are the confidence scores

TABLE I
WEIGHTING FUNCTIONS

Name	Function	FA features
GlobalInverse	$f(p,len(d)) = \frac{1}{p+1}$	FA_{GI}
GlobalLogInverse	$f(p,len(d)) = \frac{1}{log(p+2)}$	FA_{GLI}
LocalLinear	$f(p,len(d)) = \frac{len(d)-p}{len(d)}$	FA_{LL}
LocalVLinear	$f(p,len(d)) = \frac{\lvert p-\frac{len(d)-1}{2}\rvert+1}{len(d)}$	FA_{LVL}

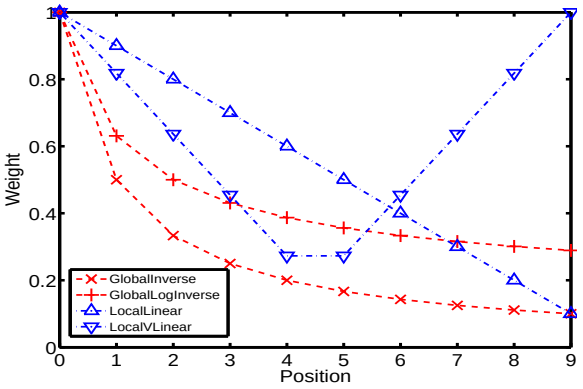


Fig. 3. The trends for different weighting functions.

which approximately indicate the probabilities that this new document belongs to each category.

Suppose there are g features, $fea_1, fea_2, ..., fea_g$, to be combined, the classifiers trained on each feature are denoted as $cla_1, cla_2, ..., cla_g$. For a given cla_i, the confidence score that a test document d belongs to the category C_j is $S_i(C_j|d)$. Thus, the final score through a combination of g classifiers is given in Eq. 11.

$$S(C_j|d) = \sum_{i=1}^{g} S_i(C_j|d)/g \tag{11}$$

V. EXPERIMENTS

SVM and kNN are two classifiers which achieved the best performance in a previous comparative study [35]. Thus, in this section, all experiments are based on these two classifiers.

A. Datasets

The Reuters-21578 corpus [19] contains 21578 articles taken from Reuters newswire[4]. The 'ModeApte' split is used. Following Yang and Liu [35], 90 categories which have at least one document in both training set and test set are extracted. After eliminating documents that don't belong to any category, there are 7770 documents in training set and 3019 documents in test set. After stemming and stop-word removal, the vocabulary contains 12158 distinct words which occur in at least two documents of the corpus.

[4]The Reuters-21578 corpus is available at http://www.daviddlewis.com /resources/testcollections/reuters21578/.

TABLE II
THE CONTINGENCY TABLE FOR CATEGORY C_i

Category C_i		Expert Judgement	
		Yes	No
Classifier Judgement	Yes	TP_i	FP_i
	No	FN_i	TN_i

TABLE III
THE GLOBAL CONTINGENCY TABLE

Category set $C = C_1, C_2, ..., C_{\|C\|}$		Expert Judgement	
		Yes	No
Classifier Judgement	Yes	$TP = \sum_{i=1}^{\|C\|} TP_i$	$FP = \sum_{i=1}^{\|C\|} FP_i$
	No	$FN = \sum_{i=1}^{\|C\|} FN_i$	$TN = \sum_{i=1}^{\|C\|} TN_i$

The 20 Newsgroup corpus [17] contains 19997 articles taken from the Usenet newsgroup collections[5]. Following Schapire and Singer [28], the duplicate documents are removed and the documents with multiple labels are detected both using the 'Xrefs' header. There are 19465 documents left. Following Bekkerman et. al. [2], four-fold cross validation is conducted which equally splits the corpus into four folds and each time uses three folds as the training set and the other fold as the test set. All 20 categories are used for experiments. After stemming and stop-word removal, the vocabulary contains 32273 distinct words which occur in at least two documents of the corpus.

The WebKB corpus [6] is a collection of 8282 web pages obtained from four academic domains[6]. Following Nigam et. al. [24], four categories: *course*, *faculty*, *project*, *student* are used and this part of corpus contains 4199 documents. Following Bekkerman et. al. [2], four-fold cross validation is conducted. All HTML tags are removed. After stop-word removal and stemming, the vocabulary contains 14467 distinct words that occur in at least two documents.

B. Performance Measure and Experimental Configuration

For evaluating the performance on these three corpus, the standard precision, recall and F1 measure is used. Given the contingency table of category C_i (Table II), the precision(p_i), recall(r_i) and F1 measure($F1_i$) of category C_i is calculated as follows.

$$p_i = \frac{TP_i}{TP_i + FP_i} \quad r_i = \frac{TP_i}{TP_i + FN_i} \quad F1_i = \frac{2 \times p_i \times r_i}{(p_i + r_i)}$$

These measures can be aggregated over all categories in two ways. One is to average each category's precision, recall and F1 to get the global precision, recall and F1. This method is called *macro-averaging*. The other is based on the global contingency table (Table III), which is called *micro-averaging*. Macro-averaging is more affected by the classifier's performance on rare categories while micro-averaging is more affected by performance on common categories. In this paper, micro-F1 and macro-F1 are both reported. Note that, for WebKB, since it is a uni-label dataset, many researchers reported accuracy on this dataset, which is the same as the micro-F1 reported in this paper[7].

The discourse passage and the window passages with different sizes are used to extract the distributional features, respectively. For Reuters-21578 and 20 Newsgroup, the paragraph is used as the discourse passage. For WebKB, since it is a web page corpus, it is difficult to obtain the discourse passage directly. Here, a web page segmentation algorithm called VIPS [3] is used to divide a web page into several blocks. The VIPS algorithm used several visual cues such as lines, blanks and colors to extract visually closely packed regions as blocks. Those blocks are used as the discourse passage for WebKB.

The kNN classifier used in this paper is similar to the one used in [35]. Specifically, a test document is assigned a score for each category according to its k nearest neighbors. Then, the score for each category is compared with the category threshold to determine whether assigning a category to this document. The category threshold is usually tuned on the training set through cross-validation. The threshold tuning method used here is Yang's *SCutFBR.1* [34] algorithm. In total, there are three parameters for kNN classifier:

- k: the number of nearest neighbors, the candidate values are: 5, 10, 20, 40, 60, 80, 100, 150, 200, 400.
- fbr: the expected minimum F1 value. If the tuned threshold outputs an F1 value less than fbr, the threshold was replaced by the highest confidence score in the training set. The candidates are eight values from 0.1 to 0.8, with a gap of 0.1.
- FeatureSize: the number of words used to index a document. The candidates are ten values from 10% to 100% of the vocabulary with a gap of 10%.

Since there are so many parameter combinations ($10 \times 8 \times 10 = 800$), parameters are chosen in a greedy way. First, fbr and FeatureSize are set as default values, which are 0.6 and 100% repectively, and k is optimized using five fold cross-validation on the training set. Second, using optimized k and default fbr, FeatureSize is optimized. Finally, fbr is optimized given the optimized k and FeatureSize. Using this method, a set of parameters are optimized for TF ('bag of words' baseline) according to micro-F1 value. Then, this set of parameters are used for the distributional features.

The SVM classifier used is LibSVM[8]. The kernel chosen is the linear kernel. All other parameters are set as default. Instances are normalized before providing to the SVM.

On these three datasets, a lot of work has been published [1], [13], [21], [26], [35], [36]. In order to make our TF ('bag of words' baseline) results comparable with previous research, the performance of term frequency reported by previous research is listed here as a reference. On Reuter-21578, Yang and Liu [35] achieved 0.8599 micro-F1 and 0.5251 macro-F1 for SVM and 0.8567 micro-F1 and 0.5242 macro-F1 for kNN and Debole and Sebastiani [7] obtained 0.86 micro-F1 and

[5]The 20 Newsgroup is available at http://kdd.ics.uci.edu/databases /20newsgroups/20newsgroups.html.

[6]The WebKB is available at: http://www-2.cs.cum.edu/afs/cs.cmu.edu /project/theo-11/www/wwkb/.

[7]For uni-label dataset, the confusion matrix is first obtained, then the contingency table for each category is calculated from the confusion matrix. After simple derivation, it is clear that on the global contingency table FN=FP=|all instances|$-$|correctly classified instances|, thus micro-F1 is equal to accuracy. $|X|$ is the number of elements in set X.

[8]The library can be found at http://www.csie.ntu.edu.tw/∼cjlin/libsvm/

TABLE IV
THE SUMMARIZATION OF THE REPORTED COMBINATIONS . FOR EACH GROUP G, G(BEST) DENOTES THE BEST PERFORMANCE OF THIS GROUP.

Group	Number of Reported Combinations	Number of Possible Combinations
TF	1: 1 (TF)	1
CP	4: 3 (3 CP features)+1 (CP(best))	$2^3 - 1 = 7$
FA	5: 4 (4 FA features)+1 (FA(best))	$2^4 - 1 = 15$
TF+CP	4: 3 (3 combinations of TF and each CP feature)+1 (TF+CP(best))	$2^3 - 1 = 7$
TF+FA	5: 4 (4 combinations of TF and each FA feature)+1 (TF+FA(best))	$2^4 - 1 = 15$
CP+FA	13: 12 (3×4 combinations of one CP feature and one FA feature)+1 (CP+FA(best))	$7 \times 15 = 105$
TF+CP+FA	13: 12 (3×4 combinations of TF, one CP feature and one FA feature)+1 (TF+CP+FA(best))	$7 \times 15 = 105$

0.52 macro-F1 for SVM. On 20 Newsgroup, Bekkerman et al. [2] achieved 0.865 micro-BEP for SVM and Nigam et. al. [24] obtained 0.82 accuracy using Naïve Bayes. On WebKB, Bekkerman et al. [2] achieved 0.923 accuracy for SVM and Nigam et. al. [24] obtained 0.81 accuracy using Naïve Bayes.

C. Effect of Distributional Features

The experiments in this section are designed to explore the effect of the distributional features. The question that we attempt to answer is: are the distributional features useful for text categorization? For 8 features (TF+ 3 CP features + 4 FA features), all 255 combinations ($2^8 - 1 = 255$) are explored. These features are organized into 7 groups: TF, CP, FA, TF+CP, TF+FA, CP+FA, TF+CP+FA. For example, all possible combinations of features from CP and features from FA form the group CP+FA. Note that to be a member of this group, the combination must contain at least one CP feature and at least one FA feature, thus the number of members of this group is $(2^3 - 1) \times (2^4 - 1) = 105$. Due to the limit of the length, the results are reported for a part of combinations of each group, which is summarized in Table IV. Note that TF is the 'bag of words' baseline.

Statistical significance tests including the micro sign test (miS) and the macro sign test (maS) are conducted. Suppose that there are a group of instances and systems A and B output a value for each instance respectively. The sign test is conducted based on the number of instances for which systems A and B output different values and the number of instances for which system A outputs a better value than system B. The micro sign test was designed to evaluate the performance at a micro level and the instance corresponds to each document-category decision; the macro sign test was designed to evaluate the performance at a macro level and the instance corresponds to the F1 value of each category. Further information on these two tests can be found in [35].

TF is used as the baseline, for which the micro F1 (miF1) and macro F1 (maF1) are reported. For other features, the gain of performance compared to the baseline is reported. Suppose the performance of ith feature (fea_i) and the baseline is $pf(fea_i)$ and $pf(base)$ respectively, the gain ($Gain$) of fea_i is calculated as Eq. 12.

$$Gain(fea_i) = \frac{pf(fea_i) - pf(base)}{pf(base)} \times 100\% \qquad (12)$$

Table V shows results of kNN and SVM on three datasets using the distributional features summarized in Table IV based on the discourse passage. The results of the statistical significance tests are also given in Table V[9].

The best performance of each group is analyzed first. Since the feature combination corresponding to the best performance of each group is not necessarily the same from dataset to dataset, this type of performance show the effect that can be obtained theoretically through the combination of our proposed distributional features. In order to facilitate reading, this type of performance is extracted from Table V and shown as the boldfaced rows of Table VI.

On Reuters-21578, CP and FA both fail to improve the baseline when they are used alone. However, when they are combined with TF or combine together, some improvement over baseline is observed especially for kNN. On 20 Newsgroup, CP and FA alone both significantly improved the baseline and FA performs a little better than CP. When they are combined with TF or combined together, better performance is observed for kNN and there is no further improvement for SVM. On WebKB, the improvement from FA dominates the result. CP also helps to improve the baseline, although not as significant as FA. When different combinations are tried, no combination could perform better than FA.

From the above analysis, the distributional features are indeed helpful for text categorization. However, for the practical use, it is inconvenient and inefficient to extract a lot of features and try all possible combinations. Thus, for the convenience of the practical use, we plan to select one CP feature and one FA feature for representation. Considering the interactions between TF, CP and FA, there are 12 $(3 \times 4 = 12)$ candidates in total, which is shown in Table VII. In order to compare these candidates, we rank their performance on each classifier-dataset-measure combination according to Table V. Since there are 2 classifiers, 3 datasets and 2 performance measures, each candidate has 12 $(2 \times 3 \times 2 = 12)$ ranks in total. The average rank for each candidate is shown in Table VII. The smaller the rank is, the better the performance is.

From Table VII, it is shown that TF+CP$_{PV}$ +FA$_{GI}$ performs the best. In order to show the gap between the selected group of features, i.e. TF, CP$_{PV}$, FA$_{GI}$, and the possible best performance, we also extract the results of different combinations of TF, CP$_{PV}$, FA$_{GI}$ from Table V and list them in Table VI to facilitate comparison.

It is shown that the performance using TF, CP$_{PV}$, FA$_{GI}$ approaches the possible best performance, especially on Newsgroup and WebKB datasets. Thus, for the rest of this paper,

[9]Note that the sign of the performance gain is not necessarily the same as the sign of the sign test results.

TABLE V

RESULTS OF THE DISTRIBUTIONAL FEATURES ON THREE DATASETS (DISCOURSE PASSAGE). ** (††) AND * (†) DENOTE THE PERFORMANCE IS SIGNIFICANTLY BETTER (WORSE) THAN THE BASELINE AT 0.01 AND 0.05 SIGNIFICANCE LEVEL, RESPECTIVELY. NO MARKERS DENOTE THE COMPARABLE PERFORMANCE. THE MARKERS ON MIF1 DENOTE THE MICRO-SIGN TEST AND THE MARKERS ON MAF1 DENOTE THE MACRO-SIGN TEST.

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
TF	0.822	0.550	0.859	0.859	0.788	0.729	0.883	0.554	0.901	0.899	0.901	0.892
CP_{PN}	0.0	-1.3	3.3**	3.3**	5.4**	9.0**	0.2	1.7	1.0**	1.0**	2.6**	2.9**
CP_{FLD}	-1.3††	-5.1†	1.3**	1.3**	4.3**	7.8**	-0.8††	-2.8†	0.1	0.1*	1.5**	1.6**
CP_{PV}	0.0	-2.7	3.1**	3.0**	6.4**	10.4**	-0.4	-2.8	0.8**	0.8**	2.6**	2.9**
CP(best)	0.8	3.1*	3.9**	3.9**	6.4**	10.4**	0.2	1.7	1.1**	1.1**	2.9**	3.3**
FA_{GI}	-2.5††	-4.7††	4.1**	4.1**	7.0**	12.4**	-0.1	-1.7	2.4**	2.4**	3.8**	4.4**
FA_{GLI}	-0.3	-2.1	5.4**	5.3**	7.3**	11.9**	-0.3	-5.3	2.3**	2.3**	3.7**	4.2**
FA_{LL}	-0.4	-0.5	4.3**	4.4**	7.2**	11.4**	-0.6	-4.2	1.6**	1.6**	3.6**	3.8**
FA_{LVL}	-1.5††	-6.3†	-0.2	-0.3†	6.0**	9.6**	-1.3††	-6.0†	1.1**	1.1**	2.7**	2.8**
FA(best)	0.0	-0.5	5.6**	5.5**	7.7**	12.9**	-0.1	-1.7	2.4**	2.4**	4.4**	4.9**
TF+CP_{PN}	0.1	-0.5	2.7**	2.6**	3.7**	6.0**	0.2	1.9	0.7**	0.7**	1.9**	2.0**
TF+CP_{FLD}	0.6	0.7	2.3**	2.3**	2.8**	4.8**	-0.1	-1.5	0.5**	0.5**	1.9**	2.0**
TF+CP_{PV}	0.7	3.0	3.0**	3.0**	4.0**	6.6**	0.4*	-1.7	0.9**	0.9**	2.4**	2.7**
TF+CP(best)	0.9	3.0	3.8**	3.8**	5.1**	8.4**	0.4*	1.9	1.0**	1.0**	2.5**	2.7**
TF+FA_{GI}	0.1	0.9	5.0**	4.9**	5.3**	8.9**	0.3	-1.9	2.1**	2.1**	3.2**	3.8**
TF+FA_{GLI}	0.7	2.3	4.8**	4.8**	5.3**	8.5**	0.4*	-2.4	1.9**	1.9**	2.8**	3.0**
TF+FA_{LL}	1.0	3.2	4.4**	4.4**	5.1**	8.1**	0.0	-2.6	1.6**	1.7**	3.1**	3.6**
TF+FA_{LVL}	0.5	1.7	2.6**	2.5**	4.1**	6.8**	0.1	-2.6	1.3**	1.3**	2.1**	2.2**
TF+FA(best)	1.6**	3.2	5.8**	5.8**	7.0**	11.1**	0.4*	-1.9	2.3**	2.3**	3.9**	4.4**
CP_{PN} +FA_{GI}	0.5	1.6	5.4**	5.3**	6.7**	11.3**	0.5*	1.7	2.3**	2.3**	3.5**	4.0**
CP_{PN} +FA_{GLI}	1.1*	1.3	5.4**	5.4**	7.1**	11.3**	0.2	0.8	2.0**	2.1**	3.2**	3.6**
CP_{PN} +FA_{LL}	1.1*	1.7	5.2**	5.2**	7.2**	11.4**	0.2	-0.4*	1.7**	1.7**	3.5**	3.9**
CP_{PN} +FA_{LVL}	0.9	2.3	3.3**	3.3**	6.0**	9.6**	-0.1	-3.6	1.6**	1.6**	3.1**	3.4**
CP_{FLD} +FA_{GI}	0.0	1.7	5.0**	4.9**	6.4**	10.9**	0.2	-1.8	2.1**	2.1**	3.3**	3.9**
CP_{FLD} +FA_{GLI}	0.3	1.1	4.8**	4.8**	6.5**	10.5**	0.3	-2.0	1.9**	1.9**	3.4**	3.9**
CP_{FLD} +FA_{LL}	0.6	1.7	4.9**	4.9**	6.5**	10.6**	0.0	-1.7	1.6**	1.6**	3.5**	3.9**
CP_{FLD} +FA_{LVL}	0.5	-0.5	3.0**	3.0**	5.9**	9.6**	-0.4	-3.0	1.5**	1.5**	3.0**	3.2**
CP_{PV} +FA_{GI}	-0.1	-0.2	5.5**	5.5**	6.9**	11.6**	0.2	-1.2	2.3**	2.3**	3.8**	4.4**
CP_{PV} +FA_{GLI}	0.2	1.0	5.4**	5.3**	7.5**	11.9**	0.0	-3.2	2.0**	2.0**	3.4**	3.8**
CP_{PV} +FA_{LL}	0.6	1.0	5.1**	5.1**	7.1**	11.2**	-0.2	-3.7	1.6**	1.7**	3.4**	3.8**
CP_{PV} +FA_{LVL}	0.2	-1.9	2.6**	2.6**	6.6**	10.2**	-0.7†	-3.9	1.4**	1.4**	3.0**	3.3**
CP+FA(best)	1.6**	4.4*	5.8**	5.8**	7.6**	12.4**	0.5*	1.7	2.4**	2.4**	4.1**	4.7**
TF+CP_{PN} +FA_{GI}	0.3	0.6	5.1**	5.1**	5.6**	9.3**	0.6**	1.3	1.8**	1.8**	3.1**	3.6**
TF+CP_{PN} +FA_{GLI}	0.3	1.5	4.9**	4.9**	5.8**	9.2**	0.6**	1.8*	1.7**	1.7**	3.0**	3.3**
TF+CP_{PN} +FA_{LL}	0.7	1.8	4.9**	4.8**	5.8**	9.3**	0.4*	-1.3	1.6**	1.6**	3.1**	3.4**
TF+CP_{PN} +FA_{LVL}	0.7	1.5*	3.5**	3.5**	5.4**	8.9**	0.4*	-2.9	1.3**	1.3**	2.7**	3.0**
TF+CP_{FLD} +FA_{GI}	0.6	1.3	4.9**	4.9**	5.3**	9.0**	0.3*	-2.6	1.6**	1.6**	3.2**	3.7**
TF+CP_{FLD} +FA_{GLI}	1.0*	2.4	4.6**	4.6**	5.4**	8.8**	0.5**	-1.5	1.5**	1.5**	2.9**	3.2**
TF+CP_{FLD} +FA_{LL}	1.4**	3.5*	4.4**	4.4**	5.4**	8.8**	0.3	-1.5	1.4**	1.5**	3.1**	3.5**
TF+CP_{FLD} +FA_{LVL}	0.8	1.0	3.3**	3.3**	5.1**	8.4**	0.2	-2.5	1.2**	1.2**	2.4**	2.7**
TF+CP_{PV} +FA_{GI}	0.8	1.9	5.4**	5.3**	6.1**	10.1**	0.5**	-2.1	1.8**	1.9**	3.3**	3.8**
TF+CP_{PV} +FA_{GLI}	1.3*	5.0**	5.0**	5.0**	5.6**	9.0**	0.3	-2.6	1.7**	1.7**	2.9**	3.2**
TF+CP_{PV} +FA_{LL}	1.6**	5.3**	5.0**	4.9**	6.0**	9.5**	0.2	-2.6	1.6**	1.6**	3.1**	3.4**
TF+CP_{PV} +FA_{LVL}	1.3*	2.6*	3.6**	3.6**	5.4**	8.6**	0.1	-2.8	1.4**	1.4**	2.8**	3.2**
TF+CP+FA(best)	1.7**	5.3**	5.9**	5.9**	7.0**	11.1**	0.6**	1.8*	2.3**	2.3**	3.7**	4.3**

TABLE VI

SIMPLIFIED RESULTS OF THE DISTRIBUTIONAL FEATURES ON THREE DATASETS (DISCOURSE PASSAGE). THE MARKERS ARE THE SAME AS TABLE V.

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
TF	0.822	0.550	0.859	0.859	0.788	0.729	0.883	0.554	0.901	0.899	0.901	0.892
CP_{PV}	0.0	-2.7	3.1**	3.0**	6.4**	10.4**	-0.4	-2.8	0.8**	0.8**	2.6**	2.9**
CP(best)	**0.8**	**3.1***	**3.9****	**3.9****	**6.4****	**10.4****	**0.2**	**1.7**	**1.1****	**1.1****	**2.9****	**3.3****
FA_{GI}	-2.5††	-4.7††	4.1**	4.1**	7.0**	12.4**	-0.1	-1.7	2.4**	2.4**	3.8**	4.4**
FA(best)	**0.0**	**-0.5**	**5.6****	**5.5****	**7.7****	**12.9****	**-0.1**	**-1.7**	**2.4****	**2.4****	**4.4****	**4.9****
TF+CP_{PV}	0.7	3.0	3.0**	3.0**	4.0**	6.6**	0.4*	-1.7	0.9**	0.9**	2.4**	2.7**
TF+CP(best)	**0.9**	**3.0**	**3.8****	**3.8****	**5.1****	**8.4****	**0.4***	**1.9**	**1.0****	**1.0****	**2.5****	**2.7****
TF+FA_{GI}	0.1	0.9	5.0**	4.9**	5.3**	8.9**	0.3	-1.9	2.1**	2.1**	3.2**	3.8**
TF+FA(best)	**1.6****	**3.2**	**5.8****	**5.8****	**7.0****	**11.1****	**0.4***	**-1.9**	**2.3****	**2.3****	**3.9****	**4.4****
CP_{PV}+FA_{GI}	-0.1	-0.2	5.5**	5.5**	6.9**	11.6**	0.2	-1.2	2.3**	2.3**	3.8**	4.4**
CP+FA(best)	**1.6****	**4.4***	**5.8****	**5.8****	**7.6****	**12.4****	**0.5***	**1.7**	**2.4****	**2.4****	**4.1****	**4.7****
TF+CP_{PV}+FA_{GI}	0.8	1.9	5.4**	5.3**	6.1**	10.1**	0.5**	-2.1	1.8**	1.9**	3.3**	3.8**
TF+CP+FA(best)	**1.7****	**5.3****	**5.9****	**5.9****	**7.0****	**11.1****	**0.6****	**1.8***	**2.3****	**2.3****	**3.7****	**4.3****

TABLE VIII
COMPARISONS WITH THE STATE-OF-THE-ART TECHNIQUES ON THREE DATASETS (DISCOURSE PASSAGE). THE MARKERS ARE THE SAME AS TABLE V

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
IB [2]	0.805	0.477	0.877	0.877	0.833	0.817	0.865	0.498	0.892	0.891	0.892	0.881
TF	2.2**	15.3**	-2.0††	-2.1††	-5.4††	-10.8††	2.1**	11.3**	0.9**	0.9**	0.9**	1.3**
CP_{PV}	2.2*	12.1**	1.0**	0.9	0.7	-1.6**	1.7**	8.2**	1.7**	1.8**	3.6**	4.2**
FA_{GI}	-0.4†	9.9	2.0**	1.9**	1.3**	0.2*	2.1**	9.4**	3.3**	3.3**	4.8**	5.8**
TF+CP_{PV}	2.9**	18.8**	1.0**	0.9	-1.5††	-4.9	2.5**	9.4**	1.8**	1.8**	3.4**	4.0**
TF+FA_{GI}	2.3*	16.4**	2.8**	2.7**	-0.3	-2.8	2.4**	9.2**	3.0**	3.0**	4.2**	5.2**
CP_{PV}+FA_{GI}	2.1	15.0**	3.4**	3.3**	1.1*	-0.5*	2.3**	9.9**	3.2**	3.2**	4.8**	5.7**
TF+CP_{PV}+FA_{GI}	3.0**	17.5**	3.2**	3.1**	0.4	-1.8	2.6**	8.9**	2.8**	2.8**	4.3**	5.1**
SI [15]	0.821	0.524	0.868	0.868	0.792	0.744	0.884	0.576	0.908	0.906	0.901	0.892
TF	0.2	5.1*	-1.0††	-1.0††	-0.4	-2.0	-0.1	-3.8	-0.8††	-0.8††	0.0	0.0
CP_{PV}	0.2	2.2*	2.0**	2.0**	6.0**	8.2**	-0.5	-6.5	0.1	0.1	2.6**	2.9**
FA_{GI}	-2.3††	0.1	3.0**	3.0**	6.6**	10.2**	-0.2	-5.5	1.6**	1.6**	3.8**	4.4**
TF+CP_{PV}	0.9	8.2**	2.0**	2.0**	3.6**	4.5**	0.3	-5.5	0.1	0.1	2.4**	2.7**
TF+FA_{GI}	0.3†	6.0*	3.9**	3.8**	4.9**	6.8**	0.2	-5.6	1.3**	1.3**	3.2**	3.8**
CP_{PV}+FA_{GI}	0.1†	4.8	4.5**	4.4**	6.4**	9.4**	0.1	-5.0	1.5**	1.5**	3.8**	4.4**
TF+CP_{PV}+FA_{GI}	0.9	7.1	4.3**	4.3**	5.6**	8.0**	0.3*	-5.9	1.1**	1.1**	3.3**	3.8**
CI [32]	0.836	0.519	0.848	0.849	0.831	0.816	0.887	0.557	0.881	0.880	0.908	0.902
TF	-1.6††	6.1	1.3**	1.2**	-5.1††	-10.7	-0.5	-0.6	2.2**	2.2**	-0.8†	-1.0
CP_{PV}	-1.6††	3.2	4.4**	4.3**	1.0*	-1.5	-0.9††	-3.4	3.0**	3.1**	1.9**	1.8**
FA_{GI}	-4.0††	1.1	5.5**	5.3**	1.5**	0.3	-0.6	-2.3	4.6**	4.7**	3.0**	3.3**
TF+CP_{PV}	-0.9††	9.3	4.4**	4.3**	-1.3†	-4.8	-0.1	-2.3	3.1**	3.1**	1.7**	1.6*
TF+FA_{GI}	-1.5††	7.1	6.4**	6.2**	0.0	-2.8	-0.2	-2.4	4.3**	4.4**	2.4**	2.7**
CP_{PV}+FA_{GI}	-1.7††	5.8	6.9**	6.8**	1.4**	-0.4	-0.3	-1.8	4.5**	4.6**	3.0**	3.3**
TF+CP_{PV}+FA_{GI}	-0.8††	8.1	6.7**	6.6**	0.7	-1.7	0.0	-2.7†	4.1**	4.1**	2.5**	2.7**

TABLE VII
AVERAGE RANK OF DIFFERENT CANDIDATES

Candidate	Average Rank
TF+CP_{PN} +FA_{GI}	4.2
TF+CP_{PN} +FA_{GLI}	5.3
TF+CP_{PN} +FA_{LL}	5.9
TF+CP_{PN} +FA_{LVL}	9.7
TF+CP_{FLD} +FA_{GI}	6.7
TF+CP_{FLD} +FA_{GLI}	7.3
TF+CP_{FLD} +FA_{LL}	6.8
TF+CP_{FLD} +FA_{LVL}	10.8
TF+CP_{PV} +FA_{GI}	**2.5**
TF+CP_{PV} +FA_{GLI}	5.0
TF+CP_{PV} +FA_{LL}	4.7
TF+CP_{PV} +FA_{LVL}	9.3

we focus on using these three features.

According to Section II, a few research has been conducted to improve the 'bag of words' representation. Therefore, besides using the TF method as the baseline, it is necessary to further compare our proposed distributional features with those state-of-the-art techniques. The state-of-the-art techniques used for comparison in our experiments include: the Information Bottleneck (IB) proposed by Bekkerman et. al. [2], which represents the techniques of using new units instead of single word to index a document; the Sentence Importance based weighted term frequency (SI) proposed by Ko et. al. [15], which represents the techniques of using new intra-document based weights instead of tf; the Confidence Interval based weighting function (CI) proposed by Soucy and Mineau et. al. [32], which represents the techniques of using new inter-document based weights instead of idf. For IB, we use the package provided by the author [10]; for SI and CI, we implement them by ourselves. In order to make a fair comparison with the distributional features, the parameters of kNN and SVM are the same as those used for Table VI. Besides the parameters of classifiers, these three state-of-the-art techniques have their own parameters such as the number of word clusters of IB and the $k1$ and $k2$ of SI ($k1$ and $k2$ are the weights used to balance the effect of two types of methods of measuring the importance of sentence). The parameters of state-of-the-art techniques are optimized through the five-fold cross validation on the training set. The results of three state-of-the-art techniques are shown in Table VIII. To facilitate comparison, we also take the results of TF and the distributional features from Table VI and incorporate them into Table VIII. Specifically, each state-of-the-art technique is used as the baseline. The results of TF and the distributional features are compared to each state-of-the-art technique to report the performance gain and the significance test results.

From Table VIII, we can see that on Reuters, IB performs significantly worse than TF. This result is not strange, since previous results of IB were reported on ten largest categories [2] while here the results are reported over 90 categories. SI and CI perform comparable to TF on miF1. For maF1, SI and CI performs worse than TF with kNN and performs better than TF with SVM. The comparisons between the distributional features and the state-of-the-art techniques are similar. On Newsgroup, IB performs significantly better than TF with kNN and even if in this case, the distributional features can bring further improvements over IB. In other cases, IB, SI and CI

[10]The package of IB is obtained from http://www.cs.technion.ac.il/∼ronb/cluster.exe.

perform comparable or slightly worse than TF and the distributional features always perform significantly better than IB, SI and CI. On WebKB, IB and CI significantly improves TF with kNN and the improvement is as high as approximately 5% on miF1 and 10% on maF1. Given such noticeable improvement, the distributional features can still achieve comparable or slightly better performance over IB and CI. In other cases, the distributional features always significantly outperforms the state-of-the-art techniques. Generally, Table VIII show that the distributional features perform comparable to the state-of-the-art techniques on Reuters and perform significantly better than the state-of-the-art techniques in most cases on Newsgroup and WebKB. It is interesting to notice that when IB, SI and CI already significantly outperform TF, some comparable or even better results can still be observed by using the distributional features, which further demonstrates the effect of the distributional features. In addition, the calculation of IB, SI and CI requires category information, thus these features have to be re-calculated when new tasks come. In contrast, since the distributional features are task independent, they can be used for new tasks without change. Note that, IB, CI and the distributional features use different sources of information, thus instead of replacing each other, all these features can be combined together, which will be explored in the future.

After reporting the results of the distributional features using the discourse passage, the window passage based distributional features are also tried. For each dataset, the maximum length among 80 percent shortest documents is extracted. Then, five window sizes are tried, from 20% to 100% of this maximum length, with a gap of 20%[11]. The influence of different passages on the performance of the distributional features is shown in Fig. 4. In these figures, the y-axis is the percentage improvement over TF and the x-axis is the window size (percent of the extracted maximum length). The performance of the discourse passage is plotted as the point corresponding to the window size of 0%. In these graphs, 'CP' corresponds to CP_{PV} and 'FA' corresponds to FA_{GI}.

Fig. 4 shows that the distributional features improve the performance of the baseline especially on 20 Newsgroup dataset and WebKB dataset, no matter which type of passage is used. Furthermore, with the increase of the length of window size, the performance of the distributional features tends to become stable. It is not strange, since the estimation of a word's distribution will be more and more coarse with the increase of the window size. In the extreme case, all words will appear in the same part.

It is noticed that the improvement brought by the distributional features is more obvious for kNN than for SVM. For kNN, the similarity measure is the most essential factor for the performance, while for SVM, there are also other factors such as margin influencing the performance. Thus, SVM is less sensitive than kNN to the definition of similarity. Since the distributional features help to measure the similarity between instances more accurately, kNN naturally benefits more from these features than SVM.

Above all, the distributional features are helpful in text categorization and combining term frequency features with the distributional features results in better performance.

D. Factors influencing the performance of distributional features

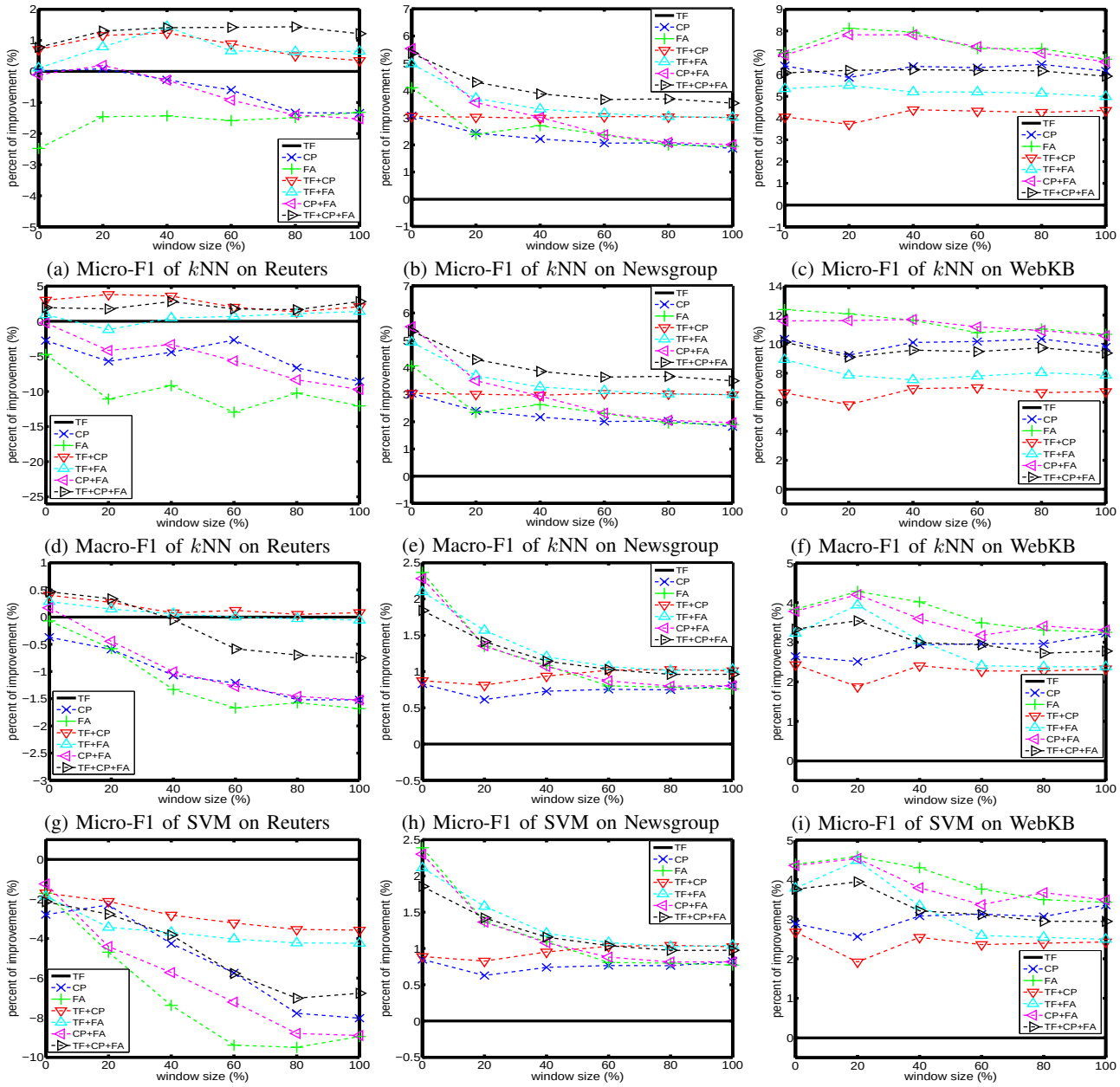
As observed, when the distributional features are introduced, there is no obvious improvement on Reuters but a significant improvement on 20 Newsgroup and WebKB. Thus, the second question arises: what factors will influence the performance of distributional features?

Recall that, when the compactness of the appearances of a word is introduced, it is assumed that a document contains several parts and the word only appears in one part is not closely related to the theme of the document. Also, when the position of the first appearance of a word is introduced, it is assumed that the word mentioned late by the author is not closely related to the theme of the document. Intuitively, these two assumptions are more likely to be satisfied when a document contains some loosely related content. Then, the following question is: in what situation may a document contain the loosely related content?

The first exploration is about the length of a document. This exploration is based on human's habit of writing. When the length of a document is limited, the author will concentrate on the most related content, such as when writing the abstract of a paper. When there is no limit for the length, the author may write some indirectly related content, such as when writing the body of a paper. The mean length of documents of the three datasets used is reported. Here, the length of a document is measured by its number of words. The average length of a document is 67.9, 115.9 and 151.7 respectively for Reuters, 20 Newsgroup and WebKB. It seems that the improvement brought by the distributional features is closely related to the mean length of documents. In order to further verify this idea, each of these three datasets is split into two new datasets, i.e. the Short dataset and the Long dataset, according to the length of documents. For each dataset, the Short dataset contains documents with length no more than 100 and the Long dataset contains documents with length more than 100. Experiments are repeated for these six new generated datasets using the discourse passage based distributional features. The results on Short and Long datasets are reported in Tables IX and X.

In order to compare the improvement on different datasets, the Relative Gain proposed in [25] is used here. Instead of directly comparing the $Gain$ over baselines, which is unfair for the improvement over a higher baseline, this method compares the ratio of the actual increase to the largest possible increase over the baseline. Suppose the performance of ith feature (fea_i) and the baseline are $pf(fea_i)$ and $pf(base)$ respectively, then the Relative Gain ($RGain$) of fea_i is calculated as Eq. 13. Note that when $pf(fea_i)$ is smaller than $pf(base)$, $RGain$ is equal to $Gain$, which can be considered as the ratio of the decrease to the largest possible decrease. The $RGain$ is calculated from the results of Tables IX and X and organized in Table XI.

[11] Since some documents are extremely longer than most documents in the dataset, if the maximum length of the whole dataset is used, the generated five window sizes will be longer than most documents, thus make little difference for most documents.



(a) Micro-F1 of kNN on Reuters (b) Micro-F1 of kNN on Newsgroup (c) Micro-F1 of kNN on WebKB

(d) Macro-F1 of kNN on Reuters (e) Macro-F1 of kNN on Newsgroup (f) Macro-F1 of kNN on WebKB

(g) Micro-F1 of SVM on Reuters (h) Micro-F1 of SVM on Newsgroup (i) Micro-F1 of SVM on WebKB

(j) Macro-F1 of SVM on Reuters (k) Macro-F1 of SVM on Newsgroup (l) Macro-F1 of SVM on WebKB

Fig. 4. Comparison of the distributional features using the discourse passage and the window passages with different sizes. X-axis denotes the window size (%) of the window passage. The zero position on X-axis corresponds to the discourse passage. Y-axis denotes the performance improvement (%) over TF.

TABLE IX

RESULTS OF THE DISTRIBUTIONAL FEATURES ON THREE SHORT DATASETS (DISCOURSE PASSAGE). THE MARKERS ARE THE SAME AS TABLE V.

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
TF	0.850	0.495	0.816	0.823	0.793	0.659	0.909	0.523	0.877	0.879	0.917	0.898
CP_{PV}	0.7	2.0	3.7**	3.5**	3.2**	2.8*	-0.3	-2.4	1.0**	1.0**	0.8**	1.1
FA_{GI}	-2.6††	-7.2††	4.8**	4.7**	2.2**	3.3*	-0.1	-3.7	2.5**	2.6**	0.3	0.6
TF+CP_{PV}	1.8**	5.5	3.6**	3.4**	2.1**	0.9*	0.2	-1.4	0.9**	0.9**	1.0**	1.5**
TF+FA_{GI}	0.8*	1.0	5.9**	5.8**	2.2**	3.1*	0.5**	-1.6	2.3**	2.4**	1.3**	1.6**
CP_{PV}+FA_{GI}	1.1**	-0.2	6.8**	6.6**	3.1**	2.8	0.1	-2.8	2.5**	2.5**	1.1**	1.7**
TF+CP_{PV}+FA_{GI}	2.0**	3.6**	6.6**	6.4**	2.4**	1.0*	0.3	-2.8	2.0**	2.1**	1.0**	1.3**

TABLE X
RESULTS OF THE DISTRIBUTIONAL FEATURES ON THREE LONG DATASETS (DISCOURSE PASSAGE). THE MARKERS ARE THE SAME AS TABLE V.

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
TF	0.635	0.395	0.868	0.849	0.752	0.711	0.733	0.364	0.908	0.901	0.863	0.860
CP_{PV}	-1.1	-4.4	3.1**	3.9**	7.8**	10.8**	-0.4	4.0	1.0**	0.9**	4.4**	4.4**
FA_{GI}	-2.1	-12.9††	3.9**	4.4**	13.2**	17.9**	1.0	-0.8	2.9**	2.8**	6.0**	6.3**
TF+CP_{PV}	1.6	-0.4*	3.3**	3.8**	4.9**	6.7**	3.0**	10.1*	1.3**	1.4**	3.3**	3.3**
TF+FA_{GI}	-0.4	-4.0	4.6**	5.2**	8.2**	11.3**	1.2*	2.6	2.7**	2.7**	4.6**	4.8**
CP_{PV}+FA_{GI}	2.4*	-3.4	5.8**	6.7**	11.5**	15.5**	2.3*	6.0	2.8**	2.8**	6.1**	6.4**
TF+CP_{PV}+FA_{GI}	0.4	-3.9	5.6**	6.5**	8.8**	12.2**	2.1**	7.2*	2.5**	2.6**	4.8**	5.0**

$$RGain(fea_i) = \begin{cases} Gain(fea_i) & \text{if } pf(fea_i) < pf(base) \\ \frac{pf(fea_i)-pf(base)}{1-pf(base)} \times 100\% & \text{otherwise} \end{cases} \quad (13)$$

According to Table XI, the distributional features brought more significant improvement on the Long dataset than on the Short dataset, although there were some exceptions indicated by '$\times$' in Table XI. It seems that the exceptions concentrate on Reuters dataset. We notice that there is a big gap between the baseline of Short part and the baseline of Long part on Reuters dataset. In this situation, comparing $RGain$ on Short and Long parts can't reflect the effect of the distributional features accurately, since the difficulty of the categorization tasks on Short and Long parts differs significantly. Note that, although $RGain$ considers such difference of difficulty to some extent, it still can't work well when the difference is big. In contrast, on Newsgroup and WebKB datasets, the baselines on Short and Long parts are comparable, thus the comparisons on these two datasets are more convincing.

Besides the length of a document, another assumption is about the writing style of a document. Note that the sources of three datasets are different: the documents in Reuters are news reports; the documents in 20 Newsgroup are newsgroup documents; the documents in WebKB are web pages. For news reports, they are written by professional journalists and editors and the writing style is formal and precise, therefore the loosely related content is less likely to appear in this type of articles. In contrast, for newsgroup documents and web pages, they are written by ordinary web users and the writing style is very casual, therefore the loosely related content is more likely to appear in this type of articles. In order to verify this assumption, the average distribution of the topical words is analyzed for three datasets. Specifically, for each dataset, the Information Gain is used to select the top 100 words as the topical words and then the average distribution of these 100 words is reported. Note that the distribution of a word is averaged over different documents where this word appears. Fig. 5 shows the average distribution of the topic words and the corresponding standard derivation over three datasets. The average distribution of the topical words is mapped to a 10 passage document. Here, 20% window passage is used[12].

[12]The fixed length passage is used to reduce the effect of the passage length on the probability that a topical word will appear. For the same reason, the last passage of a document will be ignored if it doesn't contain enough words.

Fig. 5 shows that on Reuters, the distribution of the topical words is uniform, while on Newsgroup and WebKB, the topical words are more likely to appear at the beginning of a document. These differences partly explain why the proposed distributional features perform better on Newsgroup and WebKB, especially for the dominating performance of FA on WebKB. Clearly, the writing styles account for the observed distributions of the topical words. For Reuters, since the writing style is formal, the author mentions the topic-related content all over the document; for Newsgroup and WebKB, due to the informal style, the author usually talks about the topic-related content at the start and then shifts to the loosely related content for the rest of the document.

Therefore, the answer to the second question, i.e. what factors will influence the performance of the distributional features, is: the document length and the writing style. The effect of distributional features is more obvious when the documents are long and when the writing style is informal.

E. Further analysis of the FA features

Since FA features proposed in this paper consist of two parts: the weighting function f and the strategy of only considering the first appearance of a word, it is necessary to further analyze which part brings the effect of FA features. In order to separate the influence of the weighting function, a group of weighted term frequency (WET) features are generated by using the weighting function f to weight each appearance of a word.

Suppose in a document d with n sentences, the distributional array of the word t is $array(t,d) = [c_0, c_1, ..., c_{n-1}]$. Then the WET feature is calculated as Eq. 14

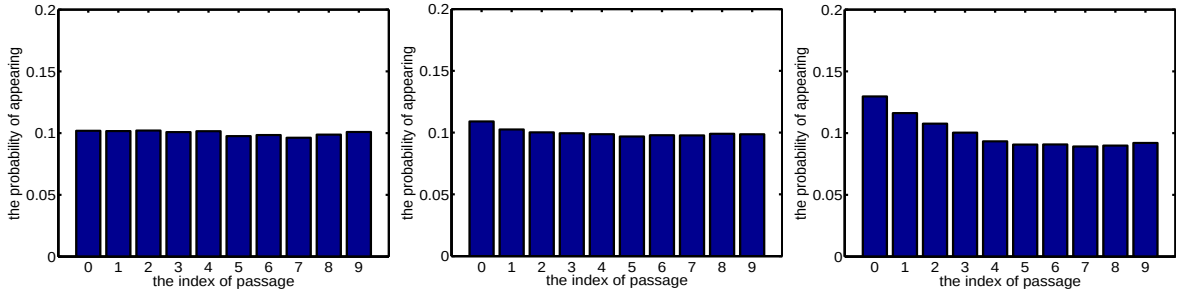
$$WET(t,d) = \frac{\sum_{i=0}^{n-1} c_i \times f(i, len(d))}{size(d)} \quad (14)$$

Corresponding to the weighting functions in Table I, four WET features can be generated, i.e. WET_{GI}, WET_{GLI}, WET_{LL} and WET_{LVL}. Since in previous section FA_{GI} is selected to represent FA features, the corresponding WET_{GI} is generated and experiments are conducted to test the effect of WET_{GI} when it is used alone and combined with TF and CP_{PV}. Table XII shows the performance of WET_{GI} using the discourse passage. In order to facilitate comparison, the performance of FA_{GI} and WET_{GI} using discourse passage is extracted from Tables VI and XII and reorganized in Table XIII.

TABLE XI

THE INFLUENCE OF DOCUMENT LENGTH ON THE RESULTS OF THE DISTRIBUTIONAL FEATURES REPORTING RELATIVE GAIN (DISCOURSE PASSAGE)

	Reuters						Newsgroup				WebKB			
	miF1			maF1			miF1		maF1		miF1		maF1	
RGain(%)	S	L		S	L		S	L	S	L	S	L	S	L
kNN														
TF	0.85	0.63		0.49	0.40		0.82	0.87	0.82	0.85	0.79	0.75	0.66	0.71
CP_{PV}	3.9	-1.1	×	2.0	-4.4	×	16.6	20.6	16.5	21.8	12.2	23.6	5.4	26.6
FA_{GI}	-2.6	-2.1		-7.2	-12.9	×	21.2	25.9	22.0	24.5	8.5	40.1	6.3	43.9
TF+CP_{PV}	10.5	2.8	×	5.4	-0.4	×	15.8	21.9	15.6	21.6	8.1	14.9	1.7	16.5
TF+FA_{GI}	4.7	-0.4	×	1.0	-4.0	×	26.1	30.5	27.0	29.3	8.5	25.1	6.1	27.9
CP_{PV}+FA_{GI}	6.0	4.1	×	-0.2	-3.4	×	30.1	37.9	30.9	37.8	12.0	34.8	5.5	38.2
TF+CP_{PV}+FA_{GI}	11.5	0.6	×	3.5	-3.9	×	29.2	36.9	29.9	36.4	9.4	26.7	1.9	30.0
SVM														
TF	0.91	0.73		0.52	0.36		0.88	0.91	0.88	0.90	0.92	0.86	0.90	0.86
CP_{PV}	-0.3	-0.4	×	-2.4	2.3		6.8	9.6	7.3	8.3	8.6	27.7	9.7	26.9
FA_{GI}	-0.1	2.9		-3.7	-0.8		17.6	29.0	18.7	25.6	3.2	37.6	4.9	38.5
TF+CP_{PV}	2.4	8.3		-1.4	5.8		6.4	12.9	6.8	13.0	11.4	20.7	13.1	20.5
TF+FA_{GI}	5.2	3.2	×	-1.6	1.5		16.5	27.2	17.3	24.5	14.1	29.2	14.1	29.7
CP_{PV}+FA_{GI}	0.8	6.3		-2.8	3.5		17.6	27.9	18.4	25.1	12.4	38.7	15.2	39.2
TF+CP_{PV}+FA_{GI}	2.7	5.7		-2.8	4.1		14.5	24.4	15.1	23.4	10.8	30.3	11.9	30.6



(a) Reuters ($std = 0.0021$) (b) Newsgroup ($std = 0.0035$) (c) WebKB ($std = 0.0137$)

Fig. 5. The average distribution of the topical words for three datasets

TABLE XII

RESULTS OF WET FEATURES ON THREE DATASETS (DISCOURSE PASSAGE). THE MARKERS ARE THE SAME AS TABLE V.

	kNN						SVM					
	Reuters		Newsgroup		WebKB		Reuters		Newsgroup		WebKB	
Gain(%)	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1	miF1	maF1
TF	0.822	0.550	0.859	0.859	0.788	0.729	0.883	0.554	0.901	0.899	0.901	0.892
CP_{PV}	0.0	-2.7	3.1**	3.0**	6.4**	10.4**	-0.4	-2.8	0.8**	0.8**	2.6**	2.9**
WET_{GI}	-1.9††	-4.4††	1.6**	1.6**	1.4**	4.7	-0.2	0.1	1.3**	1.3**	0.6*	0.7**
TF+CP_{PV}	0.7	3.0	3.0**	3.0**	4.0**	6.6**	0.4*	-1.7	0.9**	0.9**	2.4**	2.7**
TF+WET_{GI}	-0.3†	-1.3	2.6**	2.6**	1.2**	2.6**	0.3	2.4	1.3**	1.3**	0.9**	1.1**
CP_{PV}+WET_{GI}	0.7	2.8	4.4**	4.4**	5.2**	9.2**	0.4*	-2.1	2.0**	2.0**	3.1**	3.6**
TF+CP_{PV}+WET_{GI}	0.8	1.6	4.0**	4.0**	3.7**	6.6**	0.7**	-0.7*	1.5**	1.5**	2.3**	2.6**

TABLE XIII

THE COMPARISON BETWEEN THE FA FEATURE AND THE WET FEATURE WITH DISCOURSE PASSAGE REPORTING GAIN

	Reuters						Newsgroup				WebKB			
Gain(%)	miF1			maF1			miF1		maF1		miF1		maF1	
X	WET	FA		WET	FA		WET	FA	WET	FA	WET	FA	WET	FA
kNN														
X_{GI}	-1.9	-2.5	×	-4.4	-4.7	×	1.6	4.1	1.6	4.1	1.4	7.0	4.7	12.4
TF+X_{GI}	-0.3	0.1		-1.3	0.9		2.6	5.0	2.6	4.9	1.2	5.3	2.6	8.9
CP_{PV}+X_{GI}	0.7	-0.1	×	2.8	-0.2	×	4.4	5.5	4.4	5.5	5.2	6.9	9.2	11.6
TF+CP_{PV}+X_{GI}	0.8	0.8		1.6	1.9		4.0	5.4	4.0	5.3	3.7	6.1	6.6	10.1
SVM														
X_{GI}	-0.2	-0.1		0.1	-1.7	×	1.3	2.4	1.3	2.4	0.6	3.8	0.7	4.4
TF+X_{GI}	0.3	0.3		2.4	-1.9	×	1.3	2.1	1.3	2.1	0.9	3.2	1.1	3.8
CP_{PV}+X_{GI}	0.4	0.2	×	-2.1	-1.2		2.0	2.3	2.0	2.3	3.1	3.8	3.6	4.4
TF+CP_{PV}+X_{GI}	0.7	0.5	×	-0.7	-2.1	×	1.5	1.8	1.5	1.9	2.3	3.3	2.6	3.8

Table XIII shows that FA performs better than WET especially on 20 Newsgroup and WebKB. The cases where FA performs worse than WET are indicated by '$\times$'. Since WET still improves the baseline, it is believed that the effect of FA on 20 Newsgroup and WebKB is brought by both the weighting function and the aggressive strategy which throws all appearances of a word except the first one. For Reuters, the effect of this aggressive strategy is not obvious.

VI. Conclusion

Previous researches on text categorization usually use the appearance or the frequency of appearance to characterize a word. These features are not enough for fully capturing the information contained in a document. The research reported here extends a preliminary research [33] which advocates using distributional features of a word in text categorization. The distributional features encode a word's distribution from some aspects. In detail, the compactness of the appearances of a word and the position of the first appearance of a word are used. Three types of compactness-based features and four position-of-the-first-appearance-based features are implemented to reflect different considerations. A *tfidf* style equation is constructed and the ensemble learning technique is used to utilize these distributional features. Experiments show that the distributional features are useful for text categorization, especially when they are combined with term frequency or combined together. Further analysis reveals that the effect of the distributional features is obvious when the documents are long and when the writing style is informal.

Since no specific combination of TF, CP and FA consistently shows the best performance on different datasets from current experiments, how to find the optimal combination for different tasks is an important practical issue. In addition, designing the specific idf term for the distributional features is an promising direction. It is also interesting to test the effect of the distributional features on the blog dataset in the future.

Acknowledgement

We want to thank the helpful comments and suggestions from the anonymous reviewers. This research was supported by National Science Foundation of China (60505013, 60635030, 60721002) and the National High Technology Research and Development Program of China (2007AA01Z169).

References

[1] L. D. Baker and A. K. McCallum, "Distributional clustering of words for text classification," in *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval*, Melbourne, Australia, 1998, pp. 96–103.

[2] R. Bekkerman, R. El-Yaniv, N. Tishby, and Y. Winter, "Distributional word clusters vs. words for text categorization," *Journal of Machine Learning Research*, vol. 3, pp. 1182–1208, 2003.

[3] D. Cai, S.-P. Yu, J.-R. Wen, and W.-Y. Ma, "Vips: A vision-based page segmentation algorithm." Microsoft, Seattle, WA, Tech. Rep. No. MSR-TR-2003-79, 2003.

[4] J. P. Callan, "Passage retrieval evidence in document retrieval," in *Proceedings of the 17th ACM International Conference on Research and Development in Information Retrieval*, Dublin, Ireland, 1994, pp. 302–310.

[5] M. F. Caropreso, S. Matwin, and F. Sebastiani, "A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization," in *Text Databases and Document Management: Theory and Practice*, A. G. Chin, Ed. Hershey, US: Idea Group Publishing, 2001, pp. 78–102.

[6] M. Craven, D. DiPasquo, D. Freitag, A. K. McCallum, T. M. Mitchell, K. Nigam, and S. Slattery, "Learning to extract symbolic knowledge from the world wide web," in *Proceedings of the 15th National Conference for Artificial Intelligence*, Madison, WI, 1998, pp. 509–516.

[7] F. Debole and F. Sebastiani, "Supervised term weighting for automated text categorization," in *Proceedings of the 18th ACM Symposium on Applied Computing*, Melbourne, FL, 2003, pp. 784–788.

[8] T. G. Dietterich, "Machine learning research: Four current directions," *AI Magazine*, vol. 18, no. 4, pp. 97–136, 1997.

[9] S. T. Dumais, J. C. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," in *Proceedings of the 7th International Conference on Information and Knowledge Management*, Bethesda, MD, 1998, pp. 148–155.

[10] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[11] J. Fürnkranz, "A study using n-gram features for text categorization," Austrian Institute for Artificial Intelligence, Vienna, Austria, Tech. Rep. OEFAI-TR-98-30, 1998.

[12] J. Fürnkranz, T. Mitchell, and E. Riloff, "A case study in using linguistic phrases for text categorization on the WWW," in *Proceedings of the 1st AAAI Workshop on Learning for Text Categorization*, 1998, pp. 5–12.

[13] T. Joachims, "Text categorization with suport vector machines: Learning with many relevant features," in *Proceedings of the 10th European Conference on Machine Learning*, Chemnitz, Germany, 1998, pp. 137–142.

[14] J. Kim and M. H. Kim, "An evaluation of passage-based text categorization," *Journal of Intelligent Information Systems*, vol. 23, no. 1, pp. 47–65, 2004.

[15] Y. Ko, J. Park, and J. Seo, "Improving text categorization using the importance of sentences," *Information Processing and Management*, vol. 40, no. 1, pp. 65–79, 2004.

[16] M. Lan, S. Y. Sung, H. B. Low, and C. L. Tan, "A comparative study on term weighting schemes for text categorization," in *Proceedings of International Joint Conference on Neural Networks 2005*, Montreal, Canada, 2005, pp. 546–551.

[17] K. Lang, "Newsweeder: Learning to filter netnews," in *Proceedings of the 12th International Conference on Machine Learning*, Tahoe City, CA, 1995, pp. 331–339.

[18] E. Leopold and J. Kingermann, "Text categorization with support vector machines: How to represent text in input space?" *Machine Learning*, vol. 46, no. 1-3, pp. 423–444, 2002.

[19] D. Lewis, "Reuters-21578 text categorization test colleciton, dist. 1.0," 1997.

[20] D. D. Lewis, "An evaluation of phrasal and clustered representations on a text categorization task," in *Proceedings of the 15th ACM International Conference on Research and Development in Information Retrieval*, 1992, pp. 37–50.

[21] F. Li and Y. Yang., "A loss function analysis for classification methods in text categorization," in *Proceedings of the 20th International Conference on Machine Learning*, Washington DC, 2003, pp. 472–479.

[22] D. Mladenic and M. Globelnik, "Word sequences as features in text learning," in *Proceedings of the 17th Electrotechnical and Computer Science Conference*, 1998, pp. 145–148.

[23] A. Moschitti and R. Basili, "Complex linguistic features for text classification: A comprehensive study," in *Proceedings of the 26th European Conference on IR Research*, Sunderland, UK, 2004, pp. 181–196.

[24] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell, "Learning to classify text from labeled and unlabeled documents," in *Proceedings of the 15th National Conference for Artificial Intelligence*, Madison, WI, 1998, pp. 792–799.

[25] B. Raskutti, H. Ferra, and A. Kowalczyk, "Second order features for maximising text classification performance," in *Proceedings of the 12th European Conference on Machine Learning*, Freiburg, Germany, 2001, pp. 419–430.

[26] J. Rennie, L. Shih, J. Teevan, and D. Karger, "Tackling the poor assumptions of naive bayes text classifiers," in *Proceedings of the 20th International Conference on Machine Learning*, Washington DC, 2003, pp. 616–623.

[27] M. Sauban and B. Pfahringer, "Text categorization using document profiling," in *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases*, Cavtat-Dubrovnik, Croatia, 2003, pp. 411–422.

[28] R. E. Schapire and Y. Singer, "Boostexter: A boosting-based system for text categorization," *Machine Learning*, vol. 39, no. 2-3, pp. 135–168, 2000.
[29] S. Scott and S. Matwin, "Feature engineering for text classification," in *Proceedings of the 16th International Conference on Machine Learning*, Bled, Slovenia, 1999, pp. 379–388.
[30] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surverys*, vol. 34, no. 1, pp. 1–47, 2002.
[31] S. Shankar and G. Karypis, "A feature weight adjustment algorithm for document classification," in *Proceedings of SIGKDD00 Workshop on Text Mining*, Boston, MA, 2000.
[32] P. Soucy and G. W. Mineau, "Beyond tfidf weighting for text categorization in the vector space model," in *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, Edinburgh, Scotland, 2005, pp. 1130–1135.
[33] X.-B. Xue and Z.-H. Zhou, "Distributional features for text categorization," in *Proceedings of the 17th European Conference on Machine Learning*, Berlin, Germany, 2006, pp. 497–508.
[34] Y. Yang, "A study on thresholding strategies for text categorization," in *Proceedings of the 24th ACM International Conference on Research and Development in Information Retrieval*, New Orleans, LA, 2001, pp. 137–145.
[35] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd ACM International Conference on Research and Development in Information Retrieval*, Berkeley, CA, 1999, pp. 42–49.
[36] Y. Yang and J. O. Pedersen, "A comparative study on feature selectionin in text categorization," in *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, 1997, pp. 412–420.

Zhi-Hua Zhou (S'00-M'01-SM'06) received the BSc, MSc and PhD degrees in computer science from Nanjing University, China, in 1996, 1998 and 2000, respectively, all with the highest honors.

He joined the Department of Computer Science & Technology, Nanjing University, as a Lecturer in 2001, and is currently Cheung Kong Professor and Director of the LAMDA group. His research interests are in artificial intelligence, machine learning, data mining, information retrieval, pattern recognition, evolutionary computation, and neural computation. In these areas, he has published over 50 papers in leading international journals or conference proceedings.

Dr. Zhou has won various awards/honors including the National Science & Technology Award for Young Scholars of China (2006), the Award of National Science Fund for Distinguished Young Scholars of China (2003), the National Excellent Doctoral Dissertation Award of China (2003), the Microsoft Young Professorship Award (2006), etc. He is associate editor of *IEEE Transactions on Knowledge and Data Engineering*, executive editor of *Chinese Science Bulletin*, and on the editorial boards of journals including *Artificial Intelligence in Medicine*, *Intelligent Data Analysis*, *Knowledge and Information Systems*, *Science in China*, etc. He is a member of the PAKDD steering committee, program committee chair/co-chair of PAKDD'07 and PRICAI'08, vice chair or area chair of ICDM'06, ICDM'08, etc., program committee member of various international conferences including AAAI, ICML, ECML, SIGKDD, ICDM, ACM Multimedia, etc., and general chair/co-chair or program committee chair/co-chair of a dozen of native conferences. He is a senior member of the China Computer Federation (CCF) and the vice chair of the CCF Artificial Intelligence & Pattern Recognition Society, an executive committee member of the Chinese Association of Artificial Intelligence (CAAI) and the chair of the CAAI Machine Learning Society, and the chair of the IEEE Computer Society Nanjing Chapter.

Xiao-Bing Xue received his BSc and MSc degrees in computer science from Nanjing University, China, in 2004 and 2007, respectively. Currently, he is a PhD student at the Department of Computer Science of University of Massachusetts, Amherst and is a member of the CIIR Group. His research interests include information retrieval, machine learning and data mining. Currently, he works on Q&A retrieval and patent retrieval.