Context Vector Model for Document Representation: A Computational Study

Yang Wei[1,2], Jinmao Wei[1,2(✉)], and Hengpeng Xu[1,2]

[1] College of Computer and Control Engineering,
Nankai University,Weijin Rd. 94, Tianjin 300071, China
weiyang_tj@outlook.com, weijm@nankai.edu.cn,
xuhengpeng@mail.nankai.edu.cn
[2] College of Software, Nankai University, Weijin Rd. 94, Tianjin 300071, China

Abstract. To tackle the sparse data problem of the bag-of-words model for document representation, the Context Vector Model (CVM) has been proposed to enrich a document with the relatedness of all the words in a corpus to the document. The nature of CVM is the combination of word vectors, wherefore the representation method for words is essential for CVM. A computational study is performed in this paper to compare the effects of the newly proposed word representation methods embedded in CVM. The experimental results demonstrate that some of the newly proposed word representation methods significantly improve the performance of CVM, for they estimate the relatedness between words better.

Keywords: Document representation · Word vector · Relatedness

1 Introduction

Since representing documents in a feature space [21] is a pre-requisite work for many machine learning algorithms, e.g., text classification and clustering, converting a raw text to a fixed-length vector has long been studied. Perhaps the most common vector representation for texts is the bag-of-words (BOW) model due to its simplicity, comprehensibility and acceptable accuracy. However, words are assumed to be independent of each other in BOW, where relatedness actually exists. The neglect of word relatedness incurs the sparse data problem. Specifically, BOW cannot reveal the similarities between documents composed of different words. In other words, BOW has little sense about the semantic meanings of documents.

Some dimensionality reducing methods [9,11,14,23] have successfully constructed compact feature spaces. Probabilistic generative algorithms [1] and the neural probabilistic language model [15] are the outstanding artifacts in this branch. Significant improvements have been achieved with these methods. Nevertheless, the parameters, especially dimension of the space, are often difficult to be decided.

J. Li et al. (Eds.): NLPCC 2015, LNAI 9362, pp. 194–206, 2015.
DOI: 10.1007/978-3-319-25207-0_17

The Context Vector Model (CVM) [3,8,13,20], which represents documents in the same feature space of BOW, tackles the sparse data problem by considering the relatedness of all the words in a corpus to a document. This mechanism is achieved by the combination of word vectors. More precisely, the document is represented as the weighted sum of word vectors, where the weights of the word vectors are estimated based on the frequencies of the words in the document.

Representing words as vectors, the relatedness of the dimensions to a word is evaluated by their co-occurrences with the word [5,22]. Since the relatedness between words cannot be obtained directly from the dimensionality reduction methods of word representation, e.g., LDA (Latent Dirichlet Allocation) [4] and SVD (Singular Value Decomposition) [7], or the distributed representation for words, e.g., word2vec [17], the distributional representation in the feature space of BOW [6,12] is used as the statistical foundations. With the weighted sum of the distributional word vectors, the relatedness of a word to a document is obtained by the weighted sum of the relatedness of the word to the original words in the document. In this sense, the relatedness between words plays an important role for CVM. Since several word representation methods [6,12,19] have been proposed in recent years, it's meaningful to test their effects embedded in CVM for document representation. Hence a computational study on these methods is performed in this paper.

Besides, the weighted sum of word vectors has already been proved to hinder its usage on representing the semantic meanings of phrases [18]. Intuitively, the relatedness of a word to several original words in a document, which describe the same topic, may contain repeating information. Then the sum of the relatedness overestimates the relatedness of the word to the document. We propose to keep the maximum values of the quantified scores on each dimension of the word vectors to guarantee that there is no repeating information. This combination scheme of word vectors is a variance of CVM. The word representation methods embedded in different combination schemes are also compared in this paper.

The remainder of this paper is organized as follows: Sect. 2 provides the preliminaries for BOW, CVM and the word representation methods. The detailed experimental setup is described in Sect. 3. The experimental results are presented in Sect. 4. Finally, in Sect. 5, the conclusion and the direction of future work is provided.

2 Preliminaries

2.1 The BOW Model

According to BOW, the raw collection of n documents, D, must be preprocessed for vector representation. The necessary pre-operations include tokenization, to split sentences into individual tokens; stemming, a process of reducing words to their basic forms; and stopword removal. The derived words by preprocessing constitute the collection's vocabulary V. If there are m words in the vocabulary, a feature space with m-dimensions are generated. Hence a document d could be represented as:

$$\Phi_{bow} : \mathbf{d} = (c_{v_1|d}, c_{v_2|d}, \cdots, c_{v_m|d}) \in \mathbb{R}^m, \tag{1}$$

where $c_{v_x|d}$ is the occurrence times of the word v_x in d. $c_{v_x|d}$ could be the raw occurrence times of v_x in d, while it is usually re-weighted by the popular $\mathrm{tf} \cdot \mathrm{idf}$ weighting scheme:

$$c_{v_x|d} = \mathrm{tf}_{v_x|d} \cdot \mathrm{idf}_{v_x} = \frac{c_{v_x|d}}{\sum_{y=1}^{m} c_{v_y|d}} \cdot (1 + \log_2(\frac{n}{n_{v_x}})), \tag{2}$$

where n_{v_x} is the number of documents in which v_x occurs. $\mathrm{tf}_{v_x|d}$ is called the Term Frequency of v_x in d, and idf_{v_x} is the Inverse Document Frequency of v_x in the whole corpus.

2.2 Context Vector Model

Since BOW cannot figure out similar documents composed of different words, the Context Vector Model (CVM) tries to reveal the meanings of documents with a set of weighted word vectors [3,8,13,20]. $\forall v_x \in V$, its word vector is usually defined as [5,22]:

$$\mathbf{v}_x = (\frac{c_{v_x,v_1|D}}{c_{v_x|D}}, \frac{c_{v_x,v_2|D}}{c_{v_x|D}}, ..., \frac{c_{v_x,v_m|D}}{c_{v_x|D}}), \tag{3}$$

where $c_{v_x,v_y|D}$ is the co-occurrence times between v_x and v_y in the whole corpus, and $c_{v_x|D}$ is the total times v_x occurs in D. The basic assumption of word vectors is that words that occur in similar contexts tend to have similar meanings [10]. Generally, the meanings of words should be independent of the corpus size, so $c_{v_x|D}$ is introduced to give the basic context of a word [6]. The values in a word vector measure the relatedness of the dimensions to the word.

Together with all the word vectors generated by (3), an $m \times m$ matrix is obtained, which is called *context matrix*:

$$\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_m) \ . \tag{4}$$

Then the new document vector generated by CVM is:

$$\Phi_{cvm} : \mathbf{d}' = \mathbf{dV} = \left(\sum_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_1|D}}{c_{v_x|D}}, \sum_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_2|D}}{c_{v_x|D}}, \ldots, \sum_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_m|D}}{c_{v_x|D}} \right). \tag{5}$$

Instead of $\mathrm{tf} \cdot \mathrm{idf}$, $c_{v_x|d}$ will be evaluated by the simple tf scheme in this paper, because the global importance of a word in D indicated by idf has already been described in detail by the context matrix. CVM is a method which combines the local term weights (tf) and the global word relatedness together. As a result, the generated document vector is the weighted sum of the word vectors in the context matrix according to their tf weights in d. In fact, the values for each dimension are re-estimated according to their relatedness to the documents, where the relatedness of a word to the document is estimated by the weighted sum of its

relatedness to the original words in the document. The nature of CVM incurs the reasonability of the weighted sum of word vectors strategy, for the simple sum of the relatedness of a word to the original words may contain duplicate counts. Intuitively, if two terms express the same meaning in some perspective, the more informative one will cover the other one in most cases. In particular, supposing a word is related to two original words in a document, the accumulation of the relatedness of the word to the two original words will contain duplicate counts if the original words express the same meaning. So CVM is an aggressive strategy which cannot hold unless all the words in a document are independent to each other.

On the contrary, we give a conservative strategy, which takes the strongest relatedness of a word to the original words in a document as the final relatedness of the word to the document. This strategy corresponds to the following representation for a document:

$$\Phi_{crm} : \mathbf{d}' = \left(\max_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_1|D}}{c_{v_x|D}}, \max_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_2|D}}{c_{v_x|D}}, \ldots, \max_{x=1}^{m} c_{v_x|d} \frac{c_{v_x,v_m|D}}{c_{v_x|D}} \right) . \tag{6}$$

The new representation is named as Context Vector Model with the Maximum-value-aware strategy (CVMM in short), as it reserves the maximum values on each dimension of the word vectors. CVMM will hold when all the words in a document are related to each other. The weighted sum strategy used by CVM is renamed as Context Vector Model with the Accumulation strategy (CVMA). CVMA and CVMM only differ in the combination schemes of word vectors. In practice, neither of the assumptions required by CVMA or CVMM is true. CVMA will overestimate the relatedness of a word to a document, for the duplicate counts exist. While CVMM will underestimate the relatedness of a word to a document, for the non-maximum relatedness of the word to the original words is overlooked. The practical effects of CVMA and CVMM will be compared in our experiments. In the following, CVM refers to both CVMA and CVMM.

2.3 Generating the Word Vectors

According to (5) and (6), the relatedness between words plays a central role for CVM. As there are several methods to generate the word vectors in the feature space of BOW, our motivation is to embed these methods in CVM to find the best algorithm for document representation. The first word representation method to be compared is the one proposed along with CVMA [3]:

$$\frac{c_{v_x,v_y|D}}{c_{v_x|D}} = \frac{\sum_{a=1}^{n} \frac{c_{v_x|d_a}}{\sum_{z=1}^{m} c_{v_z|d_a}} \cdot \frac{c_{v_y|d_a}}{\sum_{z=1}^{m} c_{v_z|d_a}}}{\sum_{a=1}^{n} \left(\frac{c_{v_x|d_a}}{\sum_{z=1}^{m} c_{v_z|d_a}} \sum_{b=1,b\neq x}^{m} \frac{c_{v_b|d_a}}{\sum_{z=1}^{m} c_{v_z|d_a}} \right)} . \tag{7}$$

The other word representation methods are listed in Table 1, which are proposed for word representation independently [6,12,19]. The co-occurrence times and the total occurrence times used in Table 1 are usually counted with the context window method [16], where a "window", representing a span of words,

Table 1. Methods for Generating Word Vectors

Method Name	Weighting Scheme
Binary (BNR)	0 or 1
Term Frequency (TF)	$c_{v_x,v_y\|l,D} / \sum_{y=1}^{m} c_{v_x,v_y\|l,D}$
Log of TF (LTF)	$\log_2 c_{v_x,v_y\|l,D} / \log_2 \sum_{y=1}^{m} c_{v_x,v_y\|l,D}$
Log of TF-IDF (IDF)	$\log_2 c_{v_x,v_y\|l,D} / log_2 \sum_{y=1}^{m} c_{v_x,v_y\|l,D} \cdot \log_2 \frac{n}{n_{v_x}}$
Mutual Information (MI)	$\log_2 \frac{c_{v_x,v_y\|l,D}/c_D}{(\sum_{x=1}^{m} c_{v_x,v_y\|l,D}/c_D)(\sum_{y=1}^{m} c_{v_x,v_y\|l,D}/c_D)}$

is passed over the corpus being analyzed, and words within this window are recorded as co-occurring. For instance, by setting the window length to two, the fragment "the key to success, the success to $\cdots$" can be decomposed to (the key), (key to), (to success), (success the), (the success), and (success to). The co-occurrence times between "success" and "the" are two due to the appearances of (success the) and (the success). Whereas the co-occurrence times between "success" and "key" is zero for neither (success key) nor (key success) appears. In our experiments, the window length l will be set to two, which gives the strictest definition of "co-occurrence", and is the way to involve least non-related words for a target word.

The symbol $c_{v_x,v_y|l,D}$ denotes the co-occurrence times between v_x and v_y in the corpus D with the window length l, and $c_D = \sum_{x=1}^{m} \sum_{y=1}^{m} c_{v_x,v_y|l,D}$. The BNR method sets $\frac{c_{v_x,v_y|D}}{c_{v_x|D}}$ to one if $c_{v_x,v_y|l,D}$ is bigger than zero; otherwise $c_{v_x,v_y|l,D}$ is set to zero.

3 Experimental Setup

The performance of the pairwise similarity evaluation is an important index to verify the qualities of the representations for documents. Generally, with good representation, the similarities between semantically related documents should obtain high scores, while the similarities between unrelated documents should obtain low scores. This is consistent with the purpose of the clustering task that similar documents are organized into the same group, while dissimilar documents are organized into different groups. Therefore, we evaluate our algorithms on document clustering problem with the Group-average Agglomerative Hierarchical Clustering (GAHC) algorithm. The evaluation of the similarities between documents directly affects the results of GAHC, thus can reflect the qualities of the representation methods for documents.

Table 2. Characteristics of the Reuters Subsets

NO.	Topics	m	$\bar{l_d}$	$\bar{O}$	NO.	Topics	m	$\bar{l_d}$	$\bar{O}$
D_1	earn, money-supply	1452	101.54	0.126	D_2	coffee, sugar	2586	214.28	0.118
D_3	money-supply, sugar	1987	155.62	0.121	D_4	sugar, interest	2272	161.84	0.095
D_5	crude, money-supply	2460	169.64	0.107	D_6	coffee, interest	2518	186.76	0.102
D_7	money-supply, interest	1766	128.00	0.112	D_8	crude, interest	2685	175.86	0.094
D_9	trade, interest	2727	196.76	0.096	D_{10}	coffee, money-fx	2759	213.54	0.112
D_{11}	ship, sugar	2801	180.82	0.104	D_{12}	crude, sugar	2804	203.48	0.106
D_{13}	trade, sugar	2856	224.38	0.105	D_{14}	crude, money-fx	2930	202.64	0.102
D_{15}	ship, money-fx	2950	179.98	0.094	D_{16}	acq, crude	2956	176.50	0.095
D_{17}	acq, trade	3009	197.40	0.094	D_{18}	coffee, ship	3021	205.74	0.111
D_{19}	crude, coffee	3044	228.40	0.113	D_{20}	crude, trade	3229	238.40	0.102
D_{21}	all topics	3907	126.82	0.101					

3.1 Dataset

Two document collections are used in our experiments. The first is the NSF research award abstracts[1]. One hundred documents are selected randomly from the category Materials Research (MR) and Industrial Technology (IT), fifty documents per category.

The second is the twenty-one subsets extracted from Reuters[2]. The characteristics of these subsets are described in Table 2. Column *Topics* in Table 2 states the predefined categories of each subset (*all topics* of subset D_{21} means the categories acq, coffee, crude, earn, interest, money-fx, money-supply, ship, sugar, and trade are all included in this subset). Column m states the vocabulary sizes of each subset. Column $\bar{l_d}$ states the average lengths of documents in each subset. And Column $\bar{O}$ states the average overlap ratios defined in (10). There are 200 documents in $D_1 - D_{20}$, 100 per category; and five hundred documents in D_{21}, where the ratios of the number of documents in each category reserves the original ratios in the full dataset of Reuters.

All the datasets are preprocessed by removing tags, tokenizing, stemming and stopword removal. A word is considered to be stopword if its frequency in the dataset is bigger than 0.5.

3.2 Methods to Be Compared

Thirteen methods will be tested in the following experiments, namely:

1. The BOW model with the $\mathrm{tf} \cdot \mathrm{idf}$ weighting scheme, which is used as the baseline;
2. CVMA incorporated with ACP, TF, LTF, IDF, MI and BNR, respectively;
3. CVMM incorporated with ACP, TF, LTF, IDF, MI and BNR, respectively.

[1] `http://archive.ics.uci.edu/ml/datasets/NSF+Research+Award+Abstracts+1990-2003`

[2] `http://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection`

3.3 Similarity and Distance Measure

Cosine is used to quantify the pairwise similarity between two documents. For document vectors, d_i and d_j, their cosine similarity is given by:

$$\text{sim}_{\cos}(\mathbf{d}_i, \mathbf{d}_j) = \frac{\mathbf{d}_i \mathbf{d}_j^\top}{\|\mathbf{d}_i\|_2 \|\mathbf{d}_j\|_2} \ . \tag{8}$$

The distance between two documents is defined as the following accordingly:

$$\text{Dist}(\mathbf{d}_i, \mathbf{d}_j) = 1 - \text{sim}_{\cos}(\mathbf{d}_i, \mathbf{d}_j) \ . \tag{9}$$

3.4 Evaluation Metrics

Overlap Ratio. The overlap ratio is a metric to evaluate the discrimination between two vectors, which is defined as:

$$O(\mathbf{d}_i, \mathbf{d}_j) = \frac{2 * |\mathbf{d}_i \cap \mathbf{d}_j|}{|\mathbf{d}_i| + |\mathbf{d}_j|}, \tag{10}$$

where $|\mathbf{d}_i|$ is the number of dimensions with non-zero values in $\mathbf{d}_i$, and $|\mathbf{d}_i \cap \mathbf{d}_j|$ is the number of dimensions with non-zero values in both $\mathbf{d}_i$ and $\mathbf{d}_j$.

Standard Deviation. The standard deviation of the values in a word vector is defined as:

$$\text{std}(\mathbf{v}_x) = \sqrt{\sum_{y=1}^{m} \left(\frac{c_{v_x, v_y|D}}{c_{v_x|D}} - \frac{1}{m} \sum_{z=1}^{m} \frac{c_{v_x, v_z|D}}{c_{v_x|D}} \right)^2} \ . \tag{11}$$

$\mathbf{F_1}$-Score. With the index i to denote the i-th class and j to denote cluster j,

$$\text{F}_1 = \sum_i \frac{n_i}{n} \max_j \frac{2RP}{P+R}, \tag{12}$$

where n_i is the number of documents in class i, $R = \frac{n_{i,j}}{n_i}$ which is called recall in the field of Information Retrieval (IR), and $P = \frac{n_{i,j}}{n_j}$ which is called precision in IR. $n_{i,j}$ is the number of documents in both class i and cluster j [2].

Normalized Mutual Information

$$\text{NMI} = \frac{\sum_{i,j} n_{i,j} \log \frac{n n_{i,j}}{n_i n_j}}{\sqrt{(\sum_i n_i \log \frac{n_i}{n})(\sum_j n_j \log \frac{n_j}{n})}} \ . \tag{13}$$

The range of NMI is $[0, 1]$, where a value of one denotes a perfect match between clusters and reference classes [2].

3.5 Clustering Method

GAHC considers each document as a unique cluster initially and selects a pair of clusters to merge repeatedly in the merging procedure. In each turn, the pair of the most similar clusters is selected to be merged. The similarity of two clusters is calculated as the average pairwise similarities between the documents in the two clusters. The stop of the merging procedure for GAHC is achieved by predefining the target number of clusters. Specifically, the target numbers for the subsets D_1 to D_{20} range from 2 to 20. Then the number corresponding to the best F_1-score is reserved as the final number of clusters. Similarly, for dataset D_{21}, the target numbers of clusters are set in a range from 10 to 26.

4 Experimental Results

4.1 Discrimination of Document Vectors

The discrimination of document vectors generated by different models was evaluated on the NSF dataset. CVMA and CVMM were performed with ACP. Each point in the scatter diagrams of Fig. 1 represents the results of a comparison between two document vectors. The circles in the first column are the results produced by BOW; the stars in the second column represents the results of CVMA; and the plus signs in the last column represents the results of CVMM. The horizontal axis stands for the overlap ratio of two vectors, and the vertical axis stands for their similarity. The first row in Fig. 1 exhibits the results of which both documents to be compared were extracted from MR, the second row exhibits the results of which both documents were extracted from IT, and the third row illustrates the comparing results of which one document was selected from MR and the other was selected from IT.

It's shown that the overlap ratios between the document vectors generated by BOW are very low, and the document similarities wander around a small value.

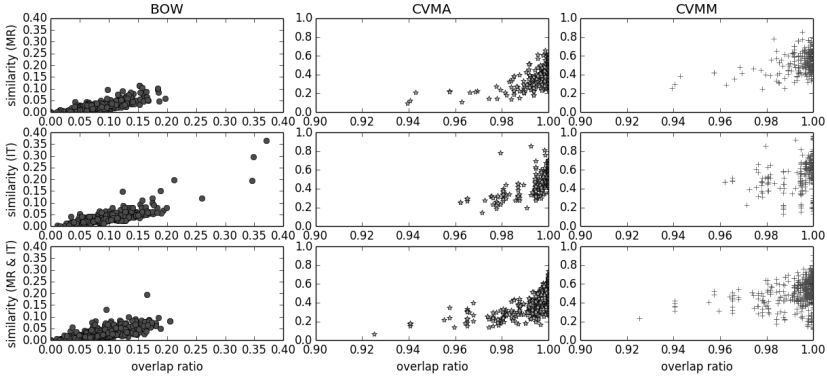

Fig. 1. Discrimination of document vectors using different representing methods.

On the contrary, the overlap ratios with CVMA or CVMM are quite high, and the document similarities range in a wide scope. This contrast reveals the advantages of using word relatedness: overcoming data sparse and magnifying the differences between document vectors. With CVM, the words which have not occurred in a document will be involved if they are related to any original words in the document. Hence the generated document vectors will have more words in common. Therefore, CVM has more smoothing power than BOW. This lead to the result that the similarities between documents composed of different words will be revealed according to the newly discovered common related words. Then similar documents are distinguished from dissimilar ones.

4.2 Performance on Document Clustering

In this experiment, the subsets of Reuters listed in Table 2 were used. On the subsets D_1 to D_{20}, the performances of the document representation methods on the variation of the data properties caused by topic changing were observed by restricting the number of categories and the number of documents, specifically, two categories and two hundred documents per subset. Since the experiment

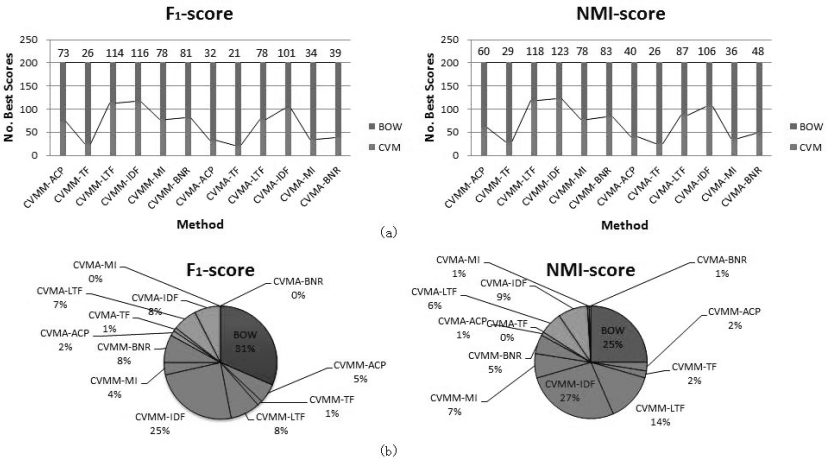

Fig. 2. Comparison between BOW and CVM on 200 datasets.

Table 3. Average Scores of the Versions of CVM on 200 Datasets

	ACP	TF	LTF	IDF	MI	BNR
F_1-score						
CVMM	0.846	0.782	0.879	0.886	0.839	0.852
CVMA	0.775	0.757	0.835	0.857	0.773	0.782
NMI-score						
CVMM	0.509	0.343	0.613	0.627	0.520	0.551
CVMA	0.341	0.275	0.486	0.561	0.334	0.358

was repeated ten times where all the documents for each subset were reselected randomly, document clustering on two hundred (20×10) subsets was performed actually. The summarized results of the document clustering are shown in Fig. 2 and Table 3. Figure 2(a) shows the numbers of the best scores each method has achieved compared with BOW, and in Fig. 2(b), the overall winning frequencies of each method on the datasets are shown. Table 3 gives the average scores of the versions of CVM on the 200 subsets. The NMI scores are the corresponding results when each method achieved their best F_1-scores.

As shown in Fig. 2, both CVMA and CVMM got competitive results compared with BOW, especially when incorporated with LTF or IDF. This agrees with our analysis that CVM obtains more smoothing power than BOW by utilizing word relatedness. However, BOW does have its particular advantages in some conditions, for BOW won 63 of 200 times in the overall competition. On subset D_{21}, the performances of BOW, CVMA and CVMM became undistinguishable. BOW got the average F_1-score of 0.625, CVMA got the average of 0.624 incorporated with ACP, and CVMM got the average of 0.636 incorporated with BNR. Since all the topics were contained in D_{21}, none of the three methods could handle the rich documents together well.

In addition, according to the relative comparison with BOW and the average scores, CVMM is superior to CVMA. This demonstrates that duplicate counts actually exist with the weighted sum strategy, and CVMM seems more plausible in practice. While the overall comparison shown in Fig. 2(b) illustrates that CVMM cannot beat CVMA all the times. It's reasonable to switch the weighted sum strategy and the maximum-value-aware strategy according to the particular dataset.

4.3 Discussion about Word Representation Methods

Six word representation methods are applied in this paper, namely, ACP, TF, LTF, IDF, MI and BNR. The experimental results in Fig. 2 and Table 3 demonstrate that the word representation methods affect the performances of CVM apparently. Instead of the different ranges of the quantified relatedness scores between words estimated by these methods, it's the relative differences among

Table 4. Correlations between Standard Deviations

	ACP	TF	LTF	IDF	MI	BNR
ACP	1.00	0.97	0.95	0.96	-0.26	0.36
TF	0.97	1.00	0.99	0.97	-0.14	0.47
LTF	0.95	0.99	1.00	0.94	-0.14	0.55
IDF	0.96	0.97	0.94	1.00	-0.13	0.31
MI	-0.26	-0.14	-0.14	-0.13	1.00	0.39
BNR	0.36	0.47	0.55	0.31	0.39	1.00

the values of the dimensions in the same vector that really matters. The standard deviation was used to evaluate the relative differences in the word vectors of the subsets D_1 to D_{20} generated with different word representation methods, respectively. The results in the same subset were averaged. The correlations between the average standard deviations corresponding to each word representation method were calculated by the Pearson Correlation[3], and the results are shown in Table 4.

It's shown that the average standard deviations with ACP, TF, LTF, and IDF have high correlations. The common input of TF, LTF and IDF is the word co-occurrence times. For a word v_x, its co-occurrence times with any dimension are smoothed by its total occurrence times with TF. The nature of ACP is the same as TF, but ACP treats words as co-occurrence as long as they appear in the same document. In LTF, the co-occurrence times are further smoothed by taking logarithm, and the co-occurrence times are smoothed one step further in IDF by introducing the inverse document frequency. Similarly, MI smooths the word co-occurrence times by taking into account the occurrence times of both the target word and the dimensions. BNR takes an extreme smoothing approach by assigning dimensions the binary scores. According to the average scores shown in Table 3, the word representation methods LTF and IDF performed consistently better than the other methods embedded in both CVMA and CVMM, which demonstrates that ACP and TF are under-smoothing policies, while MI and BNR are over-smoothing policies.

5 Conclusions

In this paper, we have compared the performance of the Context Vector Model (CVM) with the classical Bag-of-Words model (BOW) for document representation. The experimental results demonstrate that CVM has more smoothing power than BOW by considering the relatedness between words. Six representation methods for words have been embedded into CVM in our experiments; the corresponding results show that CVM severely relies on the representation methods for words. The methods incorporated with the log of term frequency and the inverse document frequency are proved to get the overall superiorities.

Besides, the combination scheme of word vectors in CVM still remains uncertain. Both the traditional weighted sum of word vectors and the proposed maximum-value-aware strategy achieve competitive results. Further study is expected to explore the inherent difference between the two combination schemes.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under grant 61070089, the Science Foundation of TianJin under grant 14JCYBJC15700.

[3] http://en.wikipedia.org/wiki/Pearson_correlation_coefficient

References

1. Anastasiu, D.C., Tagarelli, A., Karypis, G.: Document clustering: The next frontier. Tech. rep., Technical Report. University of Minnesota (2013)
2. Andrews, N.O., Fox, E.A.: Recent developments in document clustering. Computer Science, Virginia Tech, Tech Rep (2007)
3. Billhardt, H., Borrajo, D., Maojo, V.: A context vector model for information retrieval. Journal of the American Society for Information Science and Technology **53**(3), 236–249 (2002)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3**, 993–1022 (2003)
5. Blunsom, P., Grefenstette, E., Hermann, K.M., et al.: New directions in vector space models of meaning. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (2014)
6. Bullinaria, J.A., Levy, J.P.: Extracting semantic representations from word co-occurrence statistics: A computational study. Behavior Research Methods **39**(3), 510–526 (2007)
7. Bullinaria, J.A., Levy, J.P.: Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD. Behavior Research Methods **44**(3), 890–907 (2012)
8. Cheng, X., Miao, D., Wang, C., Cao, L.: Coupled term-term relation analysis for document clustering. In: The 2013 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2013)
9. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. JASIS **41**(6), 391–407 (1990)
10. Harris, Z.S.: Distributional structure. Word (1954)
11. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM (1999)
12. Iosif, E., Potamianos, A.: Unsupervised semantic similarity computation between terms using web documents. IEEE Transactions on Knowledge and Data Engineering **22**(11), 1637–1647 (2010)
13. Kalogeratos, A., Likas, A.: Text document clustering using global term context vectors. Knowledge and Information Systems **31**(3), 455–474 (2012)
14. Karypis, G., Han, E.: Concept indexing: A fast dimensionality reduction algorithm with applications to document retrieval and categorization. Tech. rep, DTIC Document (2000)
15. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: Proceedings of the 31st International Conference on Machine Learning, vol. 32, JMLR W&CP (2014)
16. Lund, K., Burgess, C.: Producing high-dimensional semantic spaces from lexical co-occurrence. Behavior Research Methods, Instruments, & Computers **28**(2), 203–208 (1996)
17. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
18. Mitchell, J., Lapata, M.: Composition in distributional models of semantics. Cognitive Science **34**(8), 1388–1429 (2010)
19. Pangos, A., Iosif, E., Potamianos, A., Fosler-Lussier, E.: Combining statistical similarity measures for automatic induction of semantic classes. In: 2005 IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 278–283. IEEE (2005)

20. Rungsawang, A.: Dsir: The first trec-7 attempt. In: TREC, pp. 366–372. Citeseer (1998)
21. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM **18**(11), 613–620 (1975)
22. Turney, P.D., Pantel, P., et al.: From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research **37**(1), 141–188 (2010)
23. Wong, S.K.M., Ziarko, W., Raghavan, V.V., Wong, P.: On modeling of information retrieval concepts in vector spaces. ACM Transactions on Database Systems (TODS) **12**(2), 299–321 (1987)