

## A FEATURE SELECTION ALGORITHM FOR DOCUMENT CLUSTERING BASED ON WORD CO-OCCURRENCE FREQUENCY

YUAN-CHAO LIU, XIAO-LONG WANG, BING-QUAN LIU

School of Computer Science and Technology, Harbin Institute of Technology, Harbin, 150001, China  
E-MAIL:lyc@insun.hit.edu.cn, wangxl@insun.hit.edu.cn, liubq@insun.hit.edu.cn

### Abstract:

Constructing feature space by only selecting more informative words can speed up document clustering algorithm greatly, and the cluster quality will not be affected. In this paper the impact of feature selection on document clustering is discussed firstly, then a new solution for feature selection was brought forward which is based on word co-occurrence frequency. According to cluster hypothesis, the documents from the same class are more similar to each other when they are represented in vector space model (VSM), so many of the words from these documents will always be in company with each other. We find these words by word co-occurrence, and then construct reduced feature space for clustering. Experiments show that the selected features are more salient. Clustering documents in the new reduced feature space, run time is shortened greatly, whereas the cluster quality is almost unchanged, thus make clustering algorithm more suitable for practical use.

### Keywords:

Document-clustering; feature selection; Cluster hypothesis; Word co-occurrence frequency

### 1. Introduction

With the explosive increase of text information on the Internet, it becomes more and more urgent to classify and organize so many unordered information to improve the information access efficiency. Recently document clustering has attracted many researchers as an important means for the organization, summarization and navigation of text information<sup>[1][2][3]</sup>. For document clustering, there is no training process, and it is not necessary to label every training document manually beforehand. Thus document clustering seems more automatic and flexible than document classification. The practical application of document clustering includes: document clustering can serve as the preprocessing step for some NLP applications such as multi-document summarization<sup>[4]</sup>; clustering the search results returned by search engine, thus make the

user find what he need in less time<sup>[5]</sup>; mine the preference model of some particular users by clustering the documents shown interested by users<sup>[6]</sup>.

Document clustering is an unsupervised learning in essence. The current document clustering algorithms include hierarchy-based clustering algorithm<sup>[7]</sup>, partition-based algorithm<sup>[8]</sup>, density-based algorithm<sup>[9]</sup> and so on. Study shows that hierarchy-based clustering algorithm is more effective. But the complexity of this kind of algorithm is  $O(n^2)$ , where  $n$  is the number of the input documents. Computing the similarity between documents is the necessary step for almost all document-clustering algorithm. Generally speaking, the computation cost is proportional to the number of the dimensions in the feature space. So we can improve the computation efficiency by reducing the feature space. In the recent time feature selection for document categorization are studied more broadly and deeply, but the literature on feature selection for document clustering are rare. The underlying reason is that for document categorization, every word can be quantified after training, thus the less salient feature words can be filtered. Whereas things are very different for document clustering, it is in essence unsupervised: there are no training documents. So it is not easy to compute the weight of every word.

In document clustering, although the feature word cannot be found directly by using class label, we can resort to the word distribution to find these words. Enlightened by this idea, a feature selection algorithm based on word co-occurrence is introduced in this paper. According to cluster hypothesis, the documents from the same class are more similar. If quantified in VSM, the possible reason that two documents are similar is that they share more words. These words can be found by finding word pairs with high co-occurrence frequency, a new reduced but still effective feature space can be constructed using these words. Experiments show that the reserved features are more informative. In the reduced space, the clustering speed is improved greatly, whereas the quality is not

affected.

## 2. Document clustering and feature selection

The mathematical model for document clustering can be depicted as follows:

Suppose  $DC = \{d_1, d_2, \dots, d_n\}$  is the document collection which contains the documents to be clustered, in VSM, every document pattern  $d_i$  can be represented as a vector in high-dimension vector space, i.e.  $d_i = \{w_1, w_2, \dots, w_l\}$ , where  $w_j$  is the weight of

document  $d_i$  in the dimension  $j$ , generally dimension is represented by every unique word from the filtered word bags which is formed by all input documents. The objective of document clustering is to partition the document collection into some subset  $C_1, C_2, \dots, C_k$ , without any prior knowledge about every category. If hard clustering, we have this formula:

$C_1 \cup C_2 \cup \dots \cup C_k = DC$ ,  $C_i \cap C_j = \Phi$ ,  $1 \leq i \neq j \leq k$ . Figure 1 shows the general preprocess steps for most of document clustering algorithm.

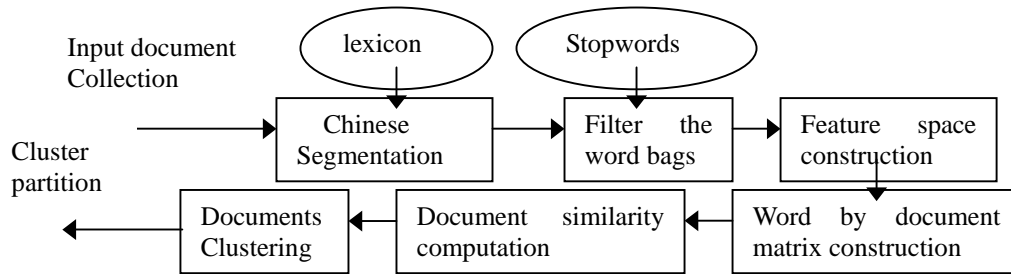


Fig. 1. the flow chart for document clustering

For most of document clustering algorithms. Similarity computation is a very crucial step. In fact, the number of dimensions can affect the cost of similarity computation greatly, at the same time the number of dimension is always proportional to the number of documents. This can be proved by cosine formula<sup>[10]</sup>, which is widely used in similarity computation:

$$sim(A, B) = \frac{\sum_{j=1}^n w_{A_j} w_{B_j}}{\sqrt{\sum_{j=1}^n w_{A_j}^2} \sqrt{\sum_{j=1}^n w_{B_j}^2}} \quad (1)$$

here  $n$  is the number of dimensions of feature space, and  $A$ 、 $B$  are two document vectors. The number of dimensions of feature space is always proportional to the number of input documents, just as figure 1 has shown. Figure 1 is the average results of 10 experiments. So we can draw a conclusion that the run time of document clustering is very sensitive to the number of input documents. This characteristics limit the practical use for many clustering algorithm.

In addition, some documents from the same class may share few words with other documents. If clustering documents in full space, these documents are less similar to

the other core documents, and these documents are generally called outliers. Outliers may be wrongly classified.

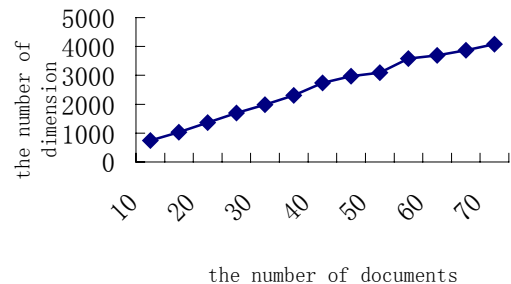


Fig.2. The relation between the number of dimension and the number of input documents

With regards to feature selection for document clustering, there is a wrap method<sup>[11]</sup> in literature. The basic idea is to evaluate the candidate feature subset using clustering algorithm. This method rely more on clustering algorithm, and lack effective clustering criterion to evaluate the cluster quality in different space. Most methods suppose that the feature words are independent to each other. In fact, most words are related to each other to some extent. Latent semantic indexing<sup>[12]</sup> can form a reduced

concept space. Every document are projected onto this new space .The study in literature has shown that clustering documents in LSI space have gotten improved quality.

### 3. Feature selection based on word co-occurrence frequency

Study shows that words that are frequent in one class whereas rare in other class are more salient in feature space. If constructing feature space using these words, the results are better. In fact, suppose the true partition is  $C_1, C_2, \dots, C_k$ , then for a particular class  $C_i$ , most words will only occur in the same class. So the feature words in the document collections are more coherent to each other. According to the knowledge of linguistics, the word co-occurrence in the same document may be random, although it purpose is to explain some topic. But if the co-occurrence frequency of these words is high in many documents, there may be strong relations between these words, and these words may be more salient in feature space. In this paper we use word co-occurrence to find these feature words and then construct reduced feature space.

Some of the terms shown in this paper are defined as follows:

**Definition 1 word co-occurrence  $f_{ij}$**  : in document collection  $DC$ , the number of documents where word  $w_i, w_j$  co-occurs;

**Definition 2 document frequency  $df(w_i)$**  :in document collection  $DC$ , the number of documents where word  $w_i$  occurs;

Word co-occurrence frequency can be gotten by algorithm 1.

#### Algorithm 1

- (1). Initialize thresholds  $\theta_A, \theta_B$ , and  $\theta_A < \theta_B$ ;
- (2). segment every documents  $d_i$  in  $DC$  and filter stop words;
- (3). construct initial feature space;
- (4). form the word by document matrix  $T$  in the initial feature space;
- (5). Find the word  $w_i$  if  $df(w_i) > \theta_A$ , record the document collection subset  $DC(w_i)$  which contains word  $w_i$ ;

(6). for every  $w_j, j \neq i$ ;

(7). compute  $f_{ij}$ , if  $f_{ij} > \theta_B$ , then record word  $w_i, w_j$  and  $f_{ij}$ ;

The word pairs with high co-occurrence frequency can be found in this way. After constructing the feature space using the words shown in pairs, then modify the word by document matrix  $T$ , remove the column by the words that are not shown in the new feature space. Thus a new word by document matrix  $T'$  can be formed. Use  $T'$  as the input of the clustering algorithm such as kmeans, then cluster the documents.

In experiment, we find there are also some high-frequency words kept in the new feature space. These words are not normal stop words and are less informative. We call these words secondary stop words. Secondary stop words can be found if their frequency is high in balanced corpus. This work can be done offline, and the statistics is loaded into memory before clustering, so the impact on the algorithm is neglectable.

The word by document matrix constructed in step 4 is  $n \times m$  matrix, Here  $n$  is the number of input documents, and the number of dimension in feature space is  $m$ .  $t_{ij}$  is the element of this matrix, representing the frequency of feature  $j$  shown in document  $i$ . Basically speaking, this matrix is a sparse matrix, as many elements are zero. The algorithm introduced in literature [13] addressed how to access this matrix effectively.

The row of the matrix is the vector of every document, i.e.  $d_j = \{tf_1, tf_2, \dots, tf_m\}$ . Only words with high frequency can be feature words candidate. In step 5 the low-frequency words are filtered.

Our experiments show that for hierarchy-based algorithm, there is always one partition that is very near to the real partition. A cluster entropy [14] based method proved to be more effective to serve as the criterion evaluation function. Our study shows that for practical use, the following strategy achieve better cluster quality and efficiency:

- (1). Reduce the feature space using algorithm 1;
- (2). Use agglomerative hierarchy-based algorithm to cluster the documents iteratively;
- (3). When the cluster number reaches a predefined value  $k$ , or when the cluster entropy reaches its minimum value, end the algorithm;
- (4). Output the cluster result.

#### 4. Experiment results and analysis

The test documents are downloaded from Internet. They are about different topics, as table 1 shows. If clustering in full space, the documents about different topic may share many words, and the documents about the same topic may share few words. Every document is segmented using Chinese maximum forward matching algorithm and filtered. The secondary stopwords file was formed by making statistics in a balanced corpus with 2 million Chinese characters. The words which frequency is higher than  $\theta_C$  ( $\theta_C = 200$ ) will be absorbed into the secondary stopwords file.

Table 1. the test document collection (Document number)

	topics	D1	D2	D3	D4
1	火星登陆 (land on the MARS)	20	---	11	---
2	总统选举 (president election)	20	20	10	10
3	利比亚核查 (nuclear inspection in Libya)	20	20	9	10
4	伊拉克战争 (Iraq war)	---	20	---	10

To evaluate our method in efficiency and quality, the experiments are carried out using the same cluster algorithm whereas in different space (full space and reduced space), the results are as follows:

The dotted line in figure 3 and figure 4 represent the run time in full space, whereas the series 2、3、4 represent the time used by feature selection、time used by clustering in reduced space and their sum. It is apparent that run time is shortened greatly for both clustering algorithm in reduced space, especially for hierarchy-based algorithm. In figure 4, the run time in reduced space is less than clustering in full space when word co-occurrence is relatively higher.

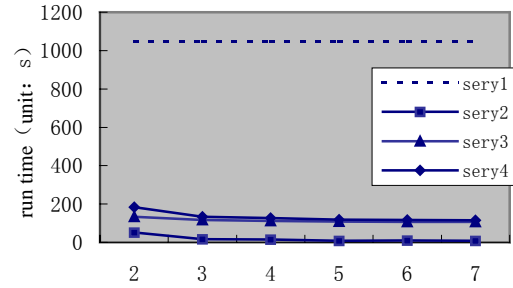


Fig. 3. the impact of feature selection on running time of hierarchy-based clustering algorithm(dataset3)

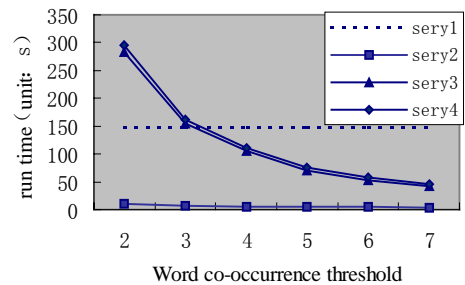


Fig.4. the impact of feature selection on running time of kmeans clustering algorithm(dataset1)

We evaluate cluster quality by F measure [15]. The definition of F value is similar to information retrieval. for a cluster  $r$  and a predefined class  $s$ ,

$$recall(r, s) = n(r, s) / n_s \quad (2)$$

$$precision(r, s) = n(r, s) / n_r \quad (3)$$

Here  $n(r, s)$  is the number of documents in the intersection of cluster  $r$  and the class  $s$ .  $n_r$  is the number of documents in cluster  $r$ ,  $n_s$  is the number of documents in class  $s$ .  $F(r, s)$  is computed as follows:

$$F(r, s) = \frac{2 * recall(r, s) * precision(r, s)}{recall(r, s) + precision(r, s)} \quad (4)$$

$$F = \sum_i \frac{n_i}{n} \max\{F(i, j)\} \quad (5)$$

Here,  $n$  is the number of all input documents, and  $i$  is one predefined class.

Table 2. the comparison of cluster quality in different space clustering using kmeans (A, B, C, D are different initial seed distribution)

Dataset	Seed Distr.	$\theta_B$	The number of feature			F measure	
			Full space	Reduced space	Compression rate	Full space	Reduced space
Dataset 1	A	2	3692	977	26.46%	1	1
		3	3692	673	18.22%	1	1
		4	3692	517	14.00%	1	1
	B	5	3692	402	10.89%	0.69	0.68
		6	3692	325	8.80%	0.69	0.68
		7	3692	269	7.29%	0.69	0.68
Dataset 2	C	2	3612	918	24.86%	1	1
		3	3612	660	18.27%	1	1
		4	3612	503	13.62%	1	1
	D	5	3612	417	11.29%	0.69	0.72
		6	3612	337	9.13%	0.69	0.70
		7	3612	278	7.70%	0.69	0.68

Table 3. the comparison of cluster quality in different feature space using hierarchy-based clustering

Dataset	$\theta_B$	The number of feature			F measure	
		Full space	Reduced space	Compression Rate.	Full space	Reduced space
Dataset 3	2	1636	324	19.80%	1	1
	3	1636	189	11.55%	1	0.86
	4	1636	117	7.15%	1	0.93
	5	1636	82	5.01%	1	0.86
Dataset 4	2	2937	734	24.99%	1	1
	3	2937	441	15.01%	1	1
	4	2937	324	11.03%	1	1
	5	2937	252	8.58%	1	1

Table 2 and table 3 are the comparison of cluster quality in different feature space. We can see that cluster quality in reduced space is very near to clustering in full space. And the quality of kmeans is sensitive to the initial seed. If the seeds are selected appropriately (see seed distribution A, C in table 2), the cluster quality will be better. Whereas the quality of hierarchy-based algorithm is relatively steady and ideal.

Figure 5 shows the relation between F measure and word co-occurrence threshold, we can see the cluster quality in reduced feature space is not sensitive to the threshold, which shows that our algorithm is effective.

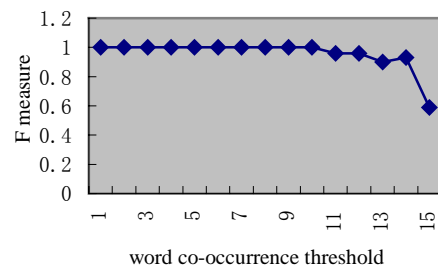


Fig. 5. the relation between F measure and word co-occurrence threshold (dataset 3, hierarchy-based cluster algorithm)

## 5. Conclusion

Clustering documents in well-reduced space can cut down the run time of the clustering algorithm dramatically, whereas the quality will not definitely be affected. A simple but very effective feature selection method is presented in this paper, which is more suitable for practical use.

## Acknowledgements

The research in this paper is supported by National Natural Science Foundation of China (60373100) and National 863 High-Tech Program of China (2002AA117010-09)

## References

- [1] Zhou haofeng, Yuan qingqing, Cheng zunping, Shi baile. PHC: A fast partition and hierarchy-based clustering algorithm. *Journal of computer and technology*. May 2003. vol.18. no.3.
- [2] Jerome Moore, Eui-Hong (Sam) Han, Daniel Boley, Maria Gini, Robert Gross, Kyle Hastings, George Karypis, Vipin Kumar, and Bamshad Mobasher Web Page Categorization and Feature Selection Using Association Rule and Principal Component Clustering, " *Decision Support Systems*, Vol. 27, No. 3, pp. 329-341, 1999.
- [3] Wu Bin, Fu Wei-Peng, Zheng Yi, Lliu Shao-Hui, and Shi Zhong-Zhi. A clustering algorithm based on swarm intelligence for web document. *Journal of computer research and development* vol.39, No.11 Nov.2002
- [4] Vasileios Hatzivassiloglou, Judith L. Klavans, Melissa L. Holcombe, Regina Barzilay, Min-Yen Kan, and Kathleen R. McKeown. SIMFINDER: A Flexible Clustering Tool for Summarization Department of Computer Science Columbia University
- [5] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey, Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections, *SIGIR '92*, Pages 318 – 329, 1992.
- [6] LIN Hong-fei, MA Ya-bin. Text filtering model based on clustering analysis *Journal of Dalian University of Technology*. Vol.42, No.2 Mar.2002
- [7] George Karypis .Eui-Hong (Sam) Han Vipin Kumar. Chameleon: A Hierarchical Clustering Algorithm Using Dynamic Modeling, *IEEE Computer, Special Issue on Data Analysis and Mining*, Vol. 32, No. 8, August 1999, pp. 68-75.
- [8] J. MacQueen. some methods for classification and analysis of multivariate observations. *Proc. 5th Berkeley Symp. Math. Statist. Prob.*, 1:281—297, 1967
- [9] Martin Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases. In *Proc. 1996*
- [10] Salton, G. Automatic Information Organization and Retrieval. New York: McGraw-Hill Press, 1968
- [11] J. G. Dy and C. E. Brodley. Visualizaion and interactive feature selection for unsupervised data .In *proc. of ACM SIGKDD*. 2000
- [12] Indexing by latent semantic analysis. Deerwester, Scott *Journal of the American Society for Information Science* 1990 41 (6) 391-407
- [13] Nazli Goharian. A Sparse Matrix Approach for information retrieval. *PHD. dissertation Florida Institute of Technology*. July 2001
- [14] Yunjae Jung. Design and Evaluation of Clustering Criterion for Optimal Hierarchical Agglomerative Clustering. *Phd. thesis. University of Minnesota*.
- [15] Michael Steinbach George Karypis. Vipin Kumar A Comparison of Document Clustering Techniques Department of Computer Science and Egeineering, University of Minnesota Technical Report #00-034