A Comparison of Document Clustering Algorithms

Yong Wang[1] and Julia Hodges[2]

Department of Computer Science & Engineering, Mississippi State University
Box 9637, Mississippi State, MS 39762

Abstract. Document clustering is a widely used strategy for information retrieval and text data mining. This paper describes the preliminary work for ongoing research of document clustering problems. A prototype of a document clustering system has been implemented and some basic aspects of document clustering problems have been studied. Our experimental results demonstrate that the average-link inter-cluster distance measure and TFIDF weighting function are good methods for the document clustering problem. Other investigators have indicated that the bisecting K-means method is the preferred method for document clustering. However, in our research we have found that, whereas the bisecting K-means method has advantages when working with large datasets, a traditional hierarchical clustering algorithm still achieves the best performance for small datasets.

1 Introduction

Data clustering partitions a set of unlabeled objects into disjoint/joint groups of clusters. In a good cluster, all the objects within a cluster are very similar while the objects in other clusters are very different. When the data processed is a set of documents, it is called document clustering. Document clustering is very important and useful in the information retrieval area. It can be applied to facilitate the retrieving the useful documents for the user. Generally, the feedback of an information retrieval system is a ranked list ordered by their estimated relevance to the query. When the volume of an information database is small and the query formulated by the user is well defined, this ranked list approach is efficient. But for a tremendous information source, such as the World Wide Web, and poor query conditions (just one or two key words), it is difficult for the retrieval system to identify the interesting items for the user. Applying documenting clustering to the retrieved documents could make it easier for the users to browse their results and locate what they want quickly. A successful example of this application is VIVISIMO (http://vivisimo.com/), which is a Web search engine that organizes search results with document clustering. Another application of document clustering is the automated or semi-automated creation of document taxonomies. A good taxonomy for Web documents is Yahoo.

This paper describes our preliminary work for research of document clustering problems. A prototype of a document clustering system has been implemented and some basic aspects of document clustering problem have been studied. In section 2, some document clustering algorithms are introduced. Section 3 presents our

Wang Y. and Hodges J. (2005).
A Comparison of Document Clustering Algorithms.
In *Proceedings of the 5th International Workshop on Pattern Recognition in Information Systems*, pages 186-191
DOI: 10.5220/0002557501860191

experimental results and analysis for different cluster distance measures, different weighting functions, and different clustering algorithms. Section 4 lists our final conclusions.

2 Documents Clustering Algorithms

Hierarchical clustering generates a hierarchical tree of clusters. Hierarchical methods can be further classified into agglomerative methods (HAC) and divisive methods. Partitioning clustering methods allocate data into a fixed number of non-empty clusters. All the clusters are in the same level. The most well-known partitioning methods following this principle are the K-means method and its variants. The buckshot method is a combination of the K-means method and HAC method. HAC method is used to select the seeds and then the K-means method is applied. The buckshot method is successfully used in a well-known document clustering system, the Scatter/Gather (SG) system [2]. The K-means method can also be used to generate hierarchical clusters. Steinbach, Karypis, and Kumar proposed bisecting K-means algorithm to generate hierarchical clusters by applying the basic K-means method recursively [7].

Besides these basic clustering algorithms, some particular algorithms for document clustering were proposed. Zamir has described the use of a suffix tree for document clustering [8]. Beil, Ester, and Xu proposed two clustering methods, FTC (Frequent Term-based Clustering) and HFTC (Hierarchical Frequent Term-based Clustering), based on frequent term sets [1]. Fung proposed another hierarchical document clustering method based on the frequent term set, HIFC (Frequent Itemset-based Hierarchical Clustering), to improve the HFTC method [3].

3 Experiments

3.1 Experimental Data and Evaluation Method

We collected 10,000 abstracts from journals belonging to ten different areas. For each area, 1000 abstracts were collected. Table 1 lists the areas and the names of the journals. This full data set was divided evenly into 5 subsets. Each subset contains 200 abstracts from each category and they are named as FDS 1 - 5. Another mini data set is selected from the full dataset. There are 1000 abstracts from 10 categories in this mini dataset. This mini dataset is partition into 5 groups evenly too. They are named as MDS 1 - 5.

All these abstracts were cut into the sentences with MXTERMINATOR [6]. Then the tokens were identified from each sentence with the Penn Treebank tokenizer. The lemmatizer in WordNet was used to convert each token into lemma [4]. All the stop words are filtered. Finally, a document is converted into a list of lemmas. These lemmas will be used to construct the feature vector for each document.

The evaluation methods F-measure and entropy, which have been used by a number of researchers, including Steinbach, Karypis, and Kumar [7], will be used in our experiments. Both entropy and F-measure are external quality measures. F-measure is a combination of precision and recall which come from information retrieval area. A higher F-measure indicates a better performance. The entropy for each cluster reflects the homogeneity of each cluster. A smaller value indicates a higher homogeneity. The detailed formula of F-measure and entropy is provided in [7].

Table 1. Journal Abstracts Data Set

Area	Journal Name	Area	Journal Name
Artificial Intelligence	Artificial Intelligence	Material	Journal of Electronic Materials
Ecology	Journal of Ecology	Nuclear	IEEE Transactions on Nuclear Science
Economy	Economic Journal	Proteomics	PubMed
History	Historical Abstracts	Sociology	Journal of Sociology
Linguistics	Journal of Linguistics	Statistics	Journal of Applied Statistics Regression Analysis

Table 2. Comparison of Inter-Cluster Distance Measures

	FDS 1	FDS 2	FDS 3	FDS 4	FDS 5	MDS 1	MDS 2	MDS 3	MDS 4	MDS 5
	F-measure									
S-link	0.19	0.23	0.25	0.19	0.19	0.52	0.35	0.40	0.37	0.50
C-link	0.18	0.18	0.18	0.18	0.18	0.18	0.18	0.18	0.18	0.19
A-link	**0.74**	**0.55**	**0.60**	**0.59**	**0.66**	**0.77**	**0.77**	**0.78**	**0.63**	**0.77**
	Entropy									
S-link	2.17	2.08	2.04	2.19	2.17	1.29	1.70	1.60	1.62	1.33
C-link	2.29	2.29	2.29	2.29	2.29	2.20	2.20	2.20	2.20	2.17
A-link	**0.73**	**1.19**	**0.98**	**1.09**	**0.91**	**0.58**	**0.51**	**0.52**	**0.84**	**0.64**

3.2 Experimental Results and Analysis

Comparison of Different Cluster Distance Measures in HAC Method

There are three generally used distance measures between clusters, single link (minimum distance), complete link (maximum distance), and average link (average distance). The comparison results of these three method are listed in table 2. In both F-measure and entropy, the average-link method achieved the best performance for all full data sets and mini data sets. This result is consistent with Steinbach, Karypis, and Kumar [7]. Different from the single-link and complete-link, the average-link measure considers every pairwise distance between two elements in two clusters and averages them. This measure reflects a global relatedness between two clusters.

Comparison of Term Weighting Methods

Different term weighting methods may be used in the vector space model. The simplest method is a binary weighting function in which a value of 1 indicates the occurrence of that term and a value of 0 indicates the absence. Another term weighting method for w_{ij} is term frequency (*tf*). In this method, each feature vector is represented by a list of occurrence frequencies of all terms. In order to avoid the effect of the varying lengths of the documents, all the occurrence frequencies should be normalized before being used. Inverse document frequency (*idf*) is a method that tries to filter those terms that occur too frequently. The most widely used term weighting method in the vector space model is *tf-idf* which is a combination of *tf* and *idf*.

Binary(Bi), Term Frequency (Tf), and Term Frequency Inverse Document Frequency (Tf-Idf) were tested in this experiment. The HAC clustering method using average-link cluster distance measure was used as the clustering method. Results of this experiment are listed in table 3. Notice that TFIDF had the best results for all data sets in both F-measure and entropy. From the introduction of these three methods given in section 3, we know that both the binary method and TF method try to weight a feature based just on the individual document. They assume that a feature with a high frequency in a document may be a key word for this document and should be useful to distinguish this document from others. This assumption is untenable when this feature occurs in most or all documents in the whole collection. The IDF measure helps to evaluate the importance of a word to the whole collection. As a combination of the TF and IDF methods, the TFIDF weighting function tries to select the features which are important for both the individual document and the whole collection.

Table 3. Comparison of Term Weighting Methods

	FDS 1	FDS 2	FDS 3	FDS 4	FDS 5	MDS 1	MDS 2	MDS 3	MDS 4	MDS 5
	F-measure									
Bi	0.27	0.23	0.30	0.36	0.19	0.51	0.65	0.44	0.43	0.31
Tf	0.27	0.23	0.31	0.44	0.18	0.73	0.64	0.53	0.55	0.55
Tf-Idf	**0.74**	**0.55**	**0.60**	**0.59**	**0.66**	**0.77**	**0.78**	**0.78**	**0.63**	**0.77**
	Entropy									
Bi	2.00	2.16	1.96	1.75	2.24	1.12	0.81	1.40	1.39	1.79
Tf	2.03	2.15	1.81	1.52	2.28	0.59	0.78	1.04	1.03	1.04
Tf-Idf	**0.73**	**1.19**	**0.98**	**1.09**	**0.91**	**0.58**	**0.51**	**0.52**	**0.84**	**0.64**

Comparison of Clustering Algorithms

Four basic clustering algorithms, K-means, buckshot, HAC, and bisecting K-means, were selected for comparison. In this experiment, K-means method, buckshot method, and bisecting K-means method are executed 20 times to alleviate the effect of a random factor. The F-measure and entropy listed here are the average values of 20 different results. In five full data sets, we found that the bisecting method outperforms all the other methods. The K-Means method and buckshot method achieve similar results. The HAC method, only in the first dataset, gets a similar result to K-means and buckshot method. But for the other four datasets, the results of the HAC method are less than that of the K-means method and buckshot method by about 10-15 percentage points. In the five mini data sets, HAC achieves the best performance. The results of K-means, buckshot, and the bisecting K-means method are similar and low.

Our results for the full data set are consistent with the results of Steinbach, Karypis, and Kumar [7]. A comprehensive analysis provided by Steinbach et al. explained that the nature of the document clustering problem is the reason for the worse performance of hierarchical approaches and the good performance of the bisecting K-means method. In all these clustering algorithms, two documents that share more common words will be considered as more similar to each other. The problem is that two documents consisting of the same set of words may be about two totally different topics. It is very possible that the nearest neighbors of a document may belong to different categories. In the HAC method, if two documents were assigned into the same group, they will always be in the same group. This assignment may be optimal in that step, but from the view of the whole partition, it may not be optimal. The HAC method just tries to get local optimality in each step with no attempt for global optimality. The advantage of the K-means method, buckshot method and bisecting K-means method is their adjusting of each cluster after each iteration. This reassignment is helpful for a global optimality. Compared with the K-means method and buckshot method, bisecting K-means can generate more evenly partitioned clusters. A balanced performance for each cluster is helpful for a higher global result.

Table 4. Comparison of Clustering Algorithms

	FDS 1	FDS 2	FDS 3	FDS 4	FDS 5	MDS 1	MDS 2	MDS 3	MDS 4	MDS 5
	F-measure									
KM	0.77	0.72	0.79	0.72	0.74	0.41	0.48	0.42	0.39	0.42
BS	0.73	0.73	0.75	0.74	0.72	0.51	0.51	0.47	0.48	0.48
HAC	0.74	0.55	0.60	0.59	0.66	**0.77**	**0.77**	**0.78**	**0.63**	**0.77**
BiKM	**0.90**	**0.85**	**0.87**	**0.86**	**0.88**	0.38	0.40	0.34	0.36	0.37
	Entropy									
KM	0.60	0.73	0.60	0.74	0.67	1.50	1.35	1.50	1.61	1.50
BS	0.67	0.70	0.65	0.72	0.71	1.28	1.24	1.38	1.37	1.34
HAC	0.73	1.19	0.98	1.09	0.91	**0.58**	**0.51**	**0.52**	**0.84**	**0.64**
BiKM	**0.40**	**0.50**	**0.46**	**0.51**	**0.45**	1.60	1.55	1.71	1.63	1.63

Then why is the HAC clustering method the best one for the mini datasets? We think the major reason is the size of the data set. There are 2000 abstracts in each full data set and they are considered as 2000 clusters in the initial step of the HAC method. We know in each iteration of the HAC method, two nearest clusters are merged together to get local optimality. This optimality may be not helpful, and may even be harmful, for the next iteration. This loop will be repeated 1990 times to get the final 10 clusters; the advantages of each step will have counteracted with each other. Since there is no global reassignment procedure in the HAC method, the final partitions cannot be improved at any steps. In our mini data set, there are only 200 abstracts in each set. The iteration was repeated only 190 times. The advantage of the optimality in each step is still higher than that of the reassignment functions in the K-means, buckshot, and bisecting methods. When we checked the detailed debugging information for the K-means, buckshot, and bisecting methods, we found that for the mini data sets, K-means, buckshot, and the bisecting method had only 1 or 2 iterations. This means that, for those small datasets, the results of K-means, buckshot, and bisecting method are similar to that of a randomly partitioning. Notice that in the full data set, the performance of the buckshot method is similar to that of the K-means method. But in the mini data set, the performance of the buckshot method is always higher than that of K-means for all five mini datasets. This demonstrates that the use of HAC for seed selection is helpful for small data sets but not for the large data sets.

There are eight data sets were used for evaluation in Steinbach, Karypis, and Kumar’s experiments [7]. The largest data set contains about 3000 documents and smallest one contains about 1000 documents. This size is similar to the size of our full data set. This also demonstrates that the good performance of bisecting K-means method achieved by in Steinbach, Karypis, and Kumar's experiments is also based on large data set.

4 Conclusions

In this paper, we described a general document clustering system prototype and discussed three ways to achieve better performance. A brief overview about different document clustering algorithms, vector weighting functions, and distance measures was provided. From our experimental results, we can below conclusions.

For the HAC clustering method, the average-link inter-clustering distance measure is better than the single-link method and complete-link method. For weighting functions in the vector space model, the TFIDF method is better than the binary method and TF method. The TFIDF method assigns a weight to a feature by combining its importance in a document and its distinguishability for whole document set. An important term may be a medium frequency word instead of a high frequency word (too common) or a low frequency word (too particular). For large data sets, the bisecting algorithm outperforms all the other methods. But for small data sets, the HAC method gets the best performance. The K-means method has a performance that is similar to that of the buckshot method for large data sets. But for small data sets, the buckshot method is better than the K-means method. The advantage of the HAC method on small data sets improves the performance of the buckshot method.

References

1. F. Beil, M. Ester, and X. Xu, "Frequent Term-Based Text Clustering," *Proc. of the 8th International Conference on Knowledge Discovery and Data Mining,* 2002.
2. D. Cutting, D. Karger, J. Pedersen, and J. Tukey, "Scatter/Gather: a Clusterbased Approach to Browsing Large Document Collection," *Proc. of the 15th ACM SIGIR Conference,* Copenhagen, Denmark, 1992, pp. 318-329.
3. B. C. M. Fung, *Hierarchical Document Clustering Using Frequent Itemsets,* Master Thesis, Dept. Computer Science, Simon Fraser University, Canada, 2002.
4. G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to WordNet: An On-Line Lexical Database," *International Journal of Lexicography,* vol. 3, no. 4, 1990, pp. 235-312.
5. A. Ratnaparkhi, "A Maximum Entropy Part-Of-Speech Tagger," *Proc. of the Empirical Methods in Natural Language Processing Conference,* University of Pennsylvania, May 1996, pp. 17-18.
6. J. C. Reynar and A. Ratnaparkhi, "A Maximum Entropy Approach to Identifying Sentence Boundaries," *Proc. of the Fifth Conference on Applied Natural Language Processing,* Washington, D.C., March 31-April 3, 1997.
7. M. Steinbach, G. Karypis, and V. Kumar, "A Comparison of Document Clustering Techniques," *KDD Workshop on Text Mining,* 2000.
8. O. Zamir, *Clustering Web Documents: A Phrase-Based Method for Group Search Engine Results,* Ph.D. dissertation, Dept. Computer Science & Engineering, Univ. of Washington, 1999.