Environment Conservation Journal 16 (SE) 133-138, 2015
ISSN 0972-3099 (Print) 2278-5124 (Online)
Abstracted and Indexed

The comparative study of text documents clustering algorithms

Mohammad Eiman Jamnezhad[1]and Reza Fattahi[2]

Received:30.06.2015 **Revised:25.07.2015** **Accepted:30.08.2015**

Abstract

Clustering is one of the most significant research area in the field of data mining and considered as an important tool in the fast developing information explosion era.Clustering systems are used more and more often in text mining, especially in analyzing texts and to extracting knowledge they contain. Data are grouped into clusters in such a way that the data of the same group are similar and those in other groups are dissimilar. It aims to minimizing intra-class similarity and maximizing inter-class dissimilarity. Clustering is useful to obtain interesting patterns and structures from a large set of data. It can be applied in many areas, namely, DNA analysis, marketing studies, web documents, and classification. This paper aims to study and compare three text documents clustering, namely, k-means, k-medoids, and SOM through F-measure.

Keywords:K-means, Bisecting K-means, k-medoids, SOM,F-measure.

Introduction

The objective of this paper is compare some algorithms, based on clustering. Clustering of text document is the process for grouping similar objects together, called cluster. Cluster shows the groups of data and a simple presentation on behalf of all objects. The presentation of the groups can be useful in teaching. Clustering text document is unsupervised learning method and well-known technique for analyzing statistical data. It has made it possible to use the same in many fields such as, machine learning, image analyzing and pattern recognizing. Researchers have presented different methods for clusters. Strong clustering of text documents should have some qualification such as scalability, obtaining a variety of features, discovery of clusters with arbitrary shape, ability to deal with crowded and fragmented data(Wanner,2004). There are different methods for clustering text document namely, partitioning methods, hierarchical methods, network-based for multi-dimensional data and clustering text document based on restrictions(Velmurugan and Santhanam). The present study compares three, text document clustering algorithms according to assessment criteria: k-means, k-medoids and SOM. Table 1 shows data collections procedures for comparing the algorithms (Steinbach et al).

Author's Address
[1]Department of Computer, Zand Institute of Higher Education, Shiraz, Iran
[2]Department of Computer, Zand Institute of Higher Education, Shiraz, Iran
E-mail: jamnezhad@zand.ac.ir

K-Means [4]

K-means is partition-based clustering method. When it is used for text clustering, all documents will be put into k clusters randomly. The basic principle of k-means for text clustering can be depicted as follows:

Input: 'N' documents to be clustered, the cluster number 'k'.

Output: 'K' clusters, and each document will be assigned to one cluster.

1) Choose k documents randomly as the initial clustering document seeds;
2) Repeat the following two steps, if the partition is stable, then go to step 5;
3) According to the mean vector of all documents in each cluster, assign each document into most similar cluster;
4) Update the mean vector of each cluster according to the document vector in it;
5) Output the generated clusters and the partition.

Bisecting K-Means [10]

This algorithm starts with a single cluster of all documents and works in the following manner:

1) Pick a cluster to split.
2) Find 2 sub-clusters using the basic K-means algorithm.

3) Repeat step 2, the bisecting step, for a fixed number of times and take the split that produces the clustering with the highest overall similarity. (For each cluster, its similarity is the average pairwise document similarity, and it is to seek minimize that sum over all clusters.)
4) Repeat steps 1, 2 and 3 until the desired number of clusters is obtained.

Data Set	Source	Documents	Classes	Words
re0	Reuters	1504	13	11465
re1	Reuters	1657	25	3758
wap	WebAce	1560	20	8460
tr31	TREC	927	7	10128
tr45	TREC	690	10	8261
fbis	TREC	2463	17	2000
la1	TREC	3204	6	31472
la2	TREC	3075	6	31472

Table 1

K-MEDOIDS

This method uses the centroid to represent the cluster and it is sensitive to outliers. It means, that a data object with an extremely large value may disrupt the distribution of data. K-medoids method overcomes this problem by using medoids to represent the cluster rather than centroid. A medoid is the most centrally located data object in a cluster. Here, k data objects are selected randomly as medoids to represent k cluster and remaining all data objects are placed in a cluster having medoid nearest (or most similar) to that data object. After processing all data objects, the new medoid is determined which can represent cluster in a better way and the entire process is repeated. Again all data objects are bound to the clusters based on the new medoids. In each iteration, medoids change their locations step by step. Or in other words, medoids move in each iteration. This process is continued until no any medoid move. As a result, *k* clusters are found represent a set of n data objects. An algorithm for this method is given below(Kohonen)

Algorithm [7]:

Input: 'k', the number of clusters to be partitioned; 'n', the number of objects.

Output: A set of '*k*' clusters that which minimizes the sum of dissimilarities of all the objects to their nearest medoid.

1) choose 'k' objects arbitrarily as the initial medoids;
2) Repeat,
   A. Assign each remaining object to the cluster with the nearest medoid;
   B. select a non-medoid object randomly;
   C. Compute the total cost of swapping old medoid object with newly selected non-medoid object.
   D. If the total cost of swapping is less than zero, then perform that swap operation to form the new set of k-medoids.
3) Until no change.

SOM

SOM (Self-Organizing feature Maps) was proposed by professor T.Kohonen(Kohonen1982). Since this process is automatic, all the input documents will be clustered. Text documents written by natural language are high-dimensional and have strong semantic features. It is hard to navigate many documents in the high-dimension space. SOM can map all these high-dimensional documents onto 2- or 1- dimensional space, and their relations in the original space can also be kept. In addition, SOM are not very sensitive to some noisy documents and the clustering quality can also be assured. Due to these advantages, SOM technology is suitable for text clustering, and has been used in many fields such as digital library(Lagus et al).

The principle of SOM for text clustering can be summarized as follows(Yiheng et al,2010)

1) Initialization. Assign some random number for all neurons and normalization. The dimension number of neuron is similar to the dimension number of all the documents;

2) Input the sample. Choose randomly one document from the document collection and send it to the SOM network;
3) Find the winner neuron. Calculate the similarity between the input document vector and the neuron vector, the neuron with the highest similarity will be the winner;
4) Adapt the vector of the winner and its neighbors. The adaptation can use the following formula:

$$m_i(t+1) = m_i(t) + \alpha(t) * h_i(t) * [x(t) - m_i(t)]$$

Where $x(t)$ is the document vector or time t, $m_i(t)$ is the original vector of neuron I, $m_i(t+1)$ is the neuron vector after adaptation. $\alpha(t)$ and $h_i(t)$ are the learning rate and neighbor rate respectively. $[x(t) - m_i(t)]$represent the distance between neuron vector and document vector.

F-measure

F-measure is a measure which combines the precision and recall ideas from information retrieval [15, 16]. It is defined as a harmonic mean of precision (P) and recall (R): [9]

$$F = \frac{2PR}{P+R}$$
$$Recall(i,j) = n_{ij}/n_i$$
$$Precision(i,j) = n_{ij}/n_j$$

Where $n_{ij}$is the number of members of class *i* in cluster *j*, n_j is the number of members of cluster *j* and n_i is the number of members of class *i*.
The F-measure of cluster *j* and class *i* is then given by: [10]

$$F(i,j) = \big(2 * Recall(i,j) * Precision(i,j)\big) / \big(Precision(i,j) + Recall(i,j)\big)$$

The F-measure is given as follows.

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

Where the max is taken over all clusters at all levels, and *n* is the number of documents.

The comparison between SOM and K-means

K-means is easy to understand and usually has low computation cost. Therefore, it has become a well-known text clustering method and used in many fields(Daniel,1998)(Cutting et al,1992). The shortcoming of k-means is that the value of 'K' must be determined beforehand and the initial document seeds need to be selected randomly. And these initial setting will have impacts on the clustering results(Ng and Han,1994).
When use k-means is used to cluster documents, some rules should be taken into consideration:

(1) After initial settings (the value of 'k' and document seeds) have been determined, the clustering results will also been determined. But the clustering results will be different if the initial settings are different;
(2) when the initial settings (the value of k and the document seeds) have been determined, suppose the clustering result of the iteration n+1 is the same as the iteration 'n', then the clustering result of iteration n + m will be the same as iteration n(m>1). Thus the variation of partition can be used as the stop criterion of clustering iteration.

The neuron number of output layer in SOM network has close relation with the class number in the input document collection.
The computation complexity of k-means is O(KlN),where 'l' is the iteration count, 'N' is the document number.
The computation complexity of k-means is SOM is O(kmN), where 'k' is the neuron number, 'm' is the training count. The computation complexity of these two methods is very near.

Experimental results

The actual performance of these two methods was compared through experiments for text clustering. The document collection in experiments has 645 documents which are about different topic. Their basic properties are listed in table 2.

Category ID	Category Description	Documentation Number	Category ID	Category Description	Documentation Number
1	Crustacea	50	9	MBA	40
2	apple	50	10	MP3	40
3	sealing	50	11	game	40
4	Lu Yongxiang(Chinese name)	40	12	Jordan (English name)	40
5	Li Guojie (Chinese name)	45	13	Tsinghua University	50
6	Digital Cameras	50	14	Tourism	50
7	Joke	50	15	Lenovo	50
8	music	50	16	Health	50

Table 2: Basic Information about Datasets

Firstly the impact of the training count on the performance of SOM is examined. Here, the training count is defined as the 'C' times the size of

input documents. For example, if there are 100 documents to be clustered and C=10, the training count should be set as 100*10=1000. The experiments show that when 'C' is low, the performance of SOM text clustering will grow quickly as the 'C' value increases. When 'C' is high, the performance is not very sensitive with 'C'value (Yiheng et al, 2010).

One clustering result of SOM has been shown in table 3. The input are the documents from 4 classes: 1, 2, 3 and 4. The topology of SOM is rectangular and the output layer includes 2*2=4 neurons, C=20, In table 2, each row represents when 'C' value is different, the number of documents which these 4 neurons TL (top left), TR (top right), DL (down left), and DR (down right) has mapped. For example, "TL=29:0:0:5" denotes that neuron TL has mapped 29 documents from class 1, 5 documents from class 4, and no documents from class 2, 3. in our experiments, It is also found that when the training count is big enough, the purity of text clustering also improves, which help improve the quality of some natural language process such as multi-document summarization, TDT(Topic detect and track) and so forth.

C	F measure	TL	TR	DL	DR
1	0.79	29:0:0:5	0:0:0:25	0:16:28:0	1:14:2:0
2--4	0.79	0:0:0:25	29:0:0:5	0:16:28:0	1:14:2:0
5--9	0.79	0:0:0:25	29:0:0:5	0:16:28:0	1:14:2:0
12	0.88	28:0:0:6	0:0:0:24	2:30:2:0	0:0:28:0
13	0.88	0:0:0:24	29:0:0:6	0:0:28:0	1:30:2:0
14	0.88	30:0:0:6	0:30:2:0	0:0:0:24	0:0:28:0
15	0.90	0:0:28:0	1:30:2:0	0:0:0:26	29:0:0:4
16	0.90	0:0:28:0	28:0:0:3	2:30:2:0	0:0:0:27
17--19	0.92	3:0:0:29	27:0:0:1	0:0:29:0	0:30:1:0
20	0.93	29:0:0:4	1:30:2:0	0:0:0:26	0:0:28:0
50	0.93	29:0:0:5	0:0:30:0	0:0:0:25	1:30:0:0
100	0.93	30:0:0:6	0:30:2:0	0:0:0:24	0:0:28:0

Table 3

As stated above, both SOM and K-means need a process of initialization. We compare if they are sensitive to the initial settings. We set the 'k' value of SOM and K-means is equal, and compare their performance when k=4 and k=9. For SOM, the topology of its output layer is 2*2=4 and 3*3=9, C=20. When the training is over, each neuron in the output layer of SOM denotes documents from one class. In table 4 and table 5 the average F-measure of 20 running of both methods is shown. SOM is not sensitive to the initial settings. Whereas the clustering results of k-means is not stable and the iteration count is also different for each running. In fact, if suitable initial document seeds can be selected, k-means will converge quickly and a better clustering quality can be achieved. As standard k-means usually select seeds randomly, the clustering quality will be affected adversely. Thus when k-means is used for text clustering, it is necessary to use some method to select suitable seeds (such as min-max principle, density-based method and so forth.)

No.	SOM (C =20)	k-means (iteration count)	No.	SOM (C =20)	k-means (iteration count)
1	0.93	0.87(7)	11	0.93	0.87(6)
2	0.93	0.65(10)	12	0.93	0.92(7)
3	0.93	0.92(7)	13	0.93	0.65(7)
4	0.93	0.67(7)	14	0.93	0.66(6)
5	0.93	0.85(10)	15	0.93	0.72(7)
6	0.93	0.92(7)	16	0.93	0.85(9)
7	0.93	0.65(2)	17	0.93	0.65(9)
8	0.93	0.65(2)	18	0.93	0.90(6)
9	0.93	0.94(6)	19	0.93	0.92(7
10	0.93	0.65(7)	20	0.93	0.87(6)

Table 4

No.	SOM (C =20)	k-means (iteration count:7-15)	No.	SOM (C =20)	k-means (iteration count:7-15)
1	0.91	0.92	11	0.91	0.88
2	0.91	0.75	12	0.91	0.75
3	0.91	0.90	13	0.91	0.75
4	0.91	0.89	14	0.91	0.98
5	0.91	0.91	15	0.91	0.88
6	0.91	0.75	16	0.91	0.75
7	0.91	0.89	17	0.91	0.75
8	0.91	0.91	18	0.91	0.93
9	0.91	0.88	19	0.91	0.88
10	0.91	0.90	20	0.91	0.75

Table 5

The experimental results also proves that when the neuron number is more than the class number of input documents, as the training of SOM tend to utilize each neuron fully, some class may be represented by more than 2 neurons. In this situation, the documents from these classes will usually be mapped onto some neighboring neurons, as shown in table 6 and table 7. In table 5, there are 3*3=9 neurons in the output layer, and there are 6 classes in the input documents. Neuron N11, N33 can represent one class respectively, whereas N21, N31 actually represent one common class. In table 6, there are 2*4=8 neurons, and input documents have 5 classes.Neuron N21 itself can represent one class. Whereas neuron N11, N12 and N22 actually represent one common class.

	Column 1	Column 2	Column 3
Line 1	0:0:30:0:0:0(N11)	15:1:0:4:2:0(N21)	14:0:0:0:0:0(N31)
Line 2	0:0:0:0:0:13(N12)	1:1:0:8:0:0(N22)	0:0:0:18:0:0(N32)
Line 3	0:12:0:0:0:17(N13)	0:16:0:0:0:0(N23)	0:0:0:0:28:0(N33)

Table 6

	Column 1	Column 2	Column 3	Column 4
Line 1	0:0:0:5:0(N11)	0:0:0:0:30(N21)	0:0:4:0:0(N31)	30:0:0:0:0:0(N41)
Line 2	0:0:0:19:0(N12)	0:0:0:6:0(N22)	0:0:26:0:0(N32)	0:30:0:0:0:0(N42)

Table 7

All these experimental results demonstrate that the topology of SOM has clear impacts on the clustering quality.

However, the clustering results of SOM can provide good navigation ability and thus makes the clustering meaningful and easy to understand. In the output layer of SOM, neighboring neurons usually maps similar documents. The documents from the same topic or similar topic will be mapped onto the same neuron or near neurons, Thus users can find the documents they need very quickly and the information access efficiency can be improved greatly. In many applications more neurons can be set (more than the possible cluster number) to cluster documents. In comparison, k-means need users to provide 'k' value to start clustering. The unsupervised property of text clustering will be affected, as in most situations, users know little about the topic structure of input documents.

The clustering quality of SOM and k-means is compared directly in some situations (the number of neurons in the output layer of SOM is same to the k value in k-means). When the output layer of SOM is 2 * 2, 2 * 3, 2 * 4 and 3 * 3, F-measure of both methods is shown in table 8. Each time 4 combinations of document class have been selected, and the mean value of 10 clustering results are utilized as the overall F measure. It can be shown that the overall clustering quality of SOM is better than K-means fully. That suggests that when the setting of output layer of SOM is reasonable, I, e, the neurons in the output layer can be used fully, SOM can achieve better clustering quality. The clustering performance of k-means is very sensitive to the initial settings, thus make its clustering quality not suitable and its F-measure less than SOM. [4]

classes	F measure		classes	F measure	
	SOM(output layer)	K-means		SOM(output layer)	K-means
1-4	0.93(2*2)	0.76	1-8	0.92(2*4)	0.78
5-8	0.91(2*2)	0.81	3-10	0.87(2*4)	0.78
9-12	0.84(2*2)	0.78	5-12	0.86(2*4)	0.81
13-16	0.93(2*2)	0.80	7-14	0.92(2*4)	0.90
1-6	0.86(2*3)	0.73	9-16	0.91(2*4)	0.83
3-9	0.92(2*3)	0.81	1-8	0.81(3*3)	0.70
6-12	0.86(2*3)	0.71	5-12	0.85(3*3)	0.80
12-16	0.87(2*3)	0.79	9-16	0.89(3*3)	0.84

Table 8

The comparison between K-means and K-medoids

K-Means:

Strengths:

- ✓ Relatively scalable and efficient in processing large data sets; complexity is O(ikn), where 'i' is the total number of iterations, 'k' is the total number of clusters, and 'n' is the total number of objects. Normally, k<<n and i<<n.
- ✓ Easy to understand and implement.

Weaknesses:

- ✓ Applicable only when the mean of a cluster is defined; not applicable to categorical data.
- ✓ Need to specify 'k', the total number of clusters in advance.
- ✓ Not suitable to discover clusters with non-convex shape, or clusters of very different size.
- ✓ Unable to handle noisy data and outliers.
- ✓ May terminate at local optimum.
- ✓ Result and total run time depends on initial partition. [13]

K-medoids

Strengths:

- ✓ More robust than k-means in the presence of noise and outliers; because a medoid is less influenced by outliers or other extreme values than a mean.

Weaknesses:

- ✓ Relatively more costly; complexity is O(ik(n-k)2), where 'i' is the total number of iterations, is the total number of clusters, and 'n' is the total number of objects.
- ✓ Relatively not so much efficient.
- ✓ Need to specify 'k', the total number of clusters in advance.
- ✓ Result and total run time depends on initial partition(Han and Kamber,2000)

Finally, the result carried out from the above study is listed in (Table 9) in the form of both clustering algorithm which highlights the realistic approach as well as desirable features of the algorithm which is useful in spatial database for different required clusters(Singh and Chauhan,2011)

Different Settings	k-means	k-medoids
Complexity	O(ikn)	O(ik(n-k)²)
Efficiency	Comparatively more	Comparatively less
Implementation	Easy	Complicated
Sensitive to Outliers?	Yes	No
Necessity of convex shape	Yes	Not so much
Advance specification of no of clusters 'k'	Required	Required
Does initial partition affects result and runtime?	Yes	Yes
Optimized for	Separated clusters	Separated clusters, Small Dataset

Table 9

Conclusions

This research tried to investigate and compare some text documents clustering algorithms. In comparison between K-means and K-medoids, it is necessary to specify the value of 'k' beforehand. The former has lower computation cost and verysensitive to crowded data. Therefore, K-medoids algorithm is better one.The experimental results have shown that K-means requires 'k' value for initial settings and is sensitive to input documents, whereas, SOM shows better results in text documents clustering of crowded data.

References

C. J. van Rijsbergen, 1989. Information Retrieval, Buttersworth, London, second edition.

Cutting, D., Karger, D., Pedersen, J. and Tukey, J. W.1992. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. Proceedings of the 15th Annual International ACM/SIGIR Conference, Copenhagen, 1992:318-329.

Daniel Boley.1998. Principal direction divisive partitioning.Data Mining and Knowledge Discovery. 1998, 2(4): 325-344.

Gerald Kowalski, Information Retrieval Systems – Theory and Implementation, Kluwer Academic Publishers, 1997.

J. Han and M. Kamber. "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, August 2000.

Jiawei Han and MichelineKamber, "Data Mining Techniques", Morgan Kaufmann Publishers, 2000.

K.Lagus,T.Honkela, S.Kaski, and T.Kohonen. (1996). Self-organizing maps of document collections: A newapproach to interactive exploration. Proceedings of the Second International Conference on KnowledgeDiscovery and Data Mining, AAAI Press, Menlo Park, California. 1996:238-243.

L. Wanner, "Introduction to Clustering Techniques",International Union of Local Authorities, (2004).

Michael Steinbach, George Karypis, Vipin Kumar, Department of Computer Science and Engineering, University of Minnesota, Technical Report #00-034

R. Ng, J. Han. (1994). Efficient and effective clustering method for spatial data mining.In Proc. of the 20th VLDB Conference, Santiago, Chile, 1994:144–155.

S. S Singh and N. C Chauhan. "K-means v/s K-medoids: A Comparative Study",BVM Engineering College and A.D.Patel Engineering College,May 2011.

Steinbach, M., Karypis, G., Kumar, V., "A Comparison of Document Clustering Techniques,"University of Minnesota, Technical Report #00-034 (2000).http://www.cs.umn.edu/tech_reports/

T. Velmurugan,and T. Santhanam, "A Survey of Partitionbased Clustering Algorithms in Data Mining: AnExperimental Approach" An experimental approach.Information. Technology.J.ournal, Vol, 10,No .3 , pp478-484,(2011).

T.Kohonen. (1982).Self-organized formation of topologically correct feature maps.Biological Cybernetics. 1982(43):59-69

Yiheng Chen,Bing Qin,Ting Liu,Yuanchao Liu,Sheng Li. (2010).The Comparison of SOM and K-means for Text Clustering.School of Computer Science and Technology,Harbin Institute of Technology PO box 321, Harbin, 150001, China

Yutaka Sasaki, Research Fellow, School of Computer Science, University of ManchesterMIB: The truth of the F-measure (2007)