Journal of Software Engineering

ISSN 1819-4311

Academic Journals Inc.

www.academicjournals.com

Journal of Software Engineering 5 (4): 136-144, 2011
ISSN 1819-4311 / DOI: 10.3923/jse.2011.136.144

Text Document Clustering Using Semantic Neighbors

Malihe Danesh and Hossein Shirgahi
Young Researchers Club, Jouybar Branch, Islamic Azad University, Jouybar, Iran

Corresponding Author: Hossein Shirgahi, Young Researchers Club, Jouybar Branch, Islamic Azad University, Jouybar, Iran

ABSTRACT

Data clustering is a powerful technique for discovering knowledge from textual documents. In this field, K-means family algorithms have many applications because of simplicity and high speed in clustering of large scale data. In these algorithms, the criterion of cosine similarity only measures the pairwise similarity of documents that it doesn't have fine operation whenever the clusters are not properly separated. On the contrary, the concepts of Neighbors and Link with the spot of general information in calculating of closeness rate of two documents, in addition to pairwise similarity between them, have better operation. In this model, semantic relations between words have been ignored and only documents with the same terms have been clustered together. This study uses WordNet Ontology for making new model of documents representation that semantic relations between words for reweighing words frequency in documents vector space model, have been used and then Neighbors and Link concepts applied to this model. Results of using the proposed method (Semantic Neighbors) on real-world text data show better operation than previous methods and more efficient in text document clustering.

Key words: Text clustering, neighbors, link, semantic similarity, cosine function

INTRODUCTION

Clustering is one of the most effective methods for extracting texts which is in the direction of the effective organization and summarization of developed documents.

Two general classifications of clustering methods are available: Agglomerative hierarchical method (Jain and Dubes, 1998) and partitional method (Li and Luo, 2008). In initialization, the algorithms of Agglomerative Hierarchical Clustering (AHC) considers each document as a cluster and use different kinds of distance function to calculate the similarity between pairs of clusters then merges them together. The step of merging is repeated so that a suitable number is obtained. In comparison with the bottom-up method of AHC, the family of k-means algorithms which belongs to partitional algorithms creates a single-surface division of documents. Every document is assigned to a cluster based on a distance criterion (between the document and each central k) after selecting k of initial center and the central k is calculated again. We repeat this step so that we can obtain an optimum collection of k cluster based on a criterion function.

Considering the high volume of operations of hierarchical algorithms on large text database, partitional clustering algorithms are used due to high quality and low computational requirements. a key feature of partitional clustering is that as criterion function is general the optimization

obtained is a product of the whole clustering process. The purpose of the criterion function is to optimize different aspects of intra-cluster similarity and inter-cluster distinction and their combinations.

A famous similarity criterion is cosine function which is widely used in the clustering algorithms of documents (Steinbach *et al.*, 2000). A cosine function can be used in k-means algorithms for assigning every document to a cluster or to the most similar cluster center. As the cosine function measures just the similarity between two documents, when clusters are not well separated they will not have suitable performance. To prevent these problems we use the concepts of neighbor and link to cluster documents (Guha *et al.*, 1999).

Lack of attention of the existing retrieval solutions to semantic similarity among words the use of just similar words among documents are among big challenges in clustering texts. We need background knowledge for overcoming this problem and enriching the existing algorithms which our source of such knowledge is ontology (Bloehdom and Hotho, 2004; Hirst and St-Onge, 1997; Hotho *et al.*, 2003; Mao and Chu, 2002; Jurisica *et al.*, 2004). Ontology is a description of concepts and relationships which was defined for sharing knowledge.

In this study, we used semantic relations among words in weighing words frequency again in the vector space model of documents and then applied the concepts of neighbor and link on the obtained model and studied its performance on k-means algorithms in different aspects. The results obtained from applying the proposed method on a set of real data show that its performance is very suitable in relation to previous methods.

The issue of clustering documents is defined as follows: we plan to divide a collection of documents into a predetermined collection of clusters in a way that the documents assigned to each cluster have the most similarity compared with the documents assigned to other different clusters. In other words, the documents of a cluster present a similar issue and the documents of different clusters present different issues.

Text clustering

Vector space model: In most document clustering algorithms, we put forward vector space model using documents (Van Rijsbergen, 1979). In this model every document d is considered as a vector in a word space and presented with word frequency vector:

$$d_{tf} = [tf_1, tf_2, ..., tf_D] \tag{1}$$

We can define center vector c_j using a set C_j of documents and its corresponding vector as follows:

$$c_j = \frac{1}{|C_j|}\sum_{d_i \in C_j} d_i \tag{2}$$

Cosine similarity criterion: One of the most famous criteria in obtaining the similarity among documents is cosine function (Steinbach *et al.*, 2000) which its value is zero when the two documents are completely similar and is one if there is no similarity and in other cases the value is between zero and one. If we suppose the two documents d_i and d_j it can be calculated as follows:

$$\cos(d_i, d_j) = \frac{d_i . d_j}{\| d_i \| \| d_j \|} \tag{3}$$

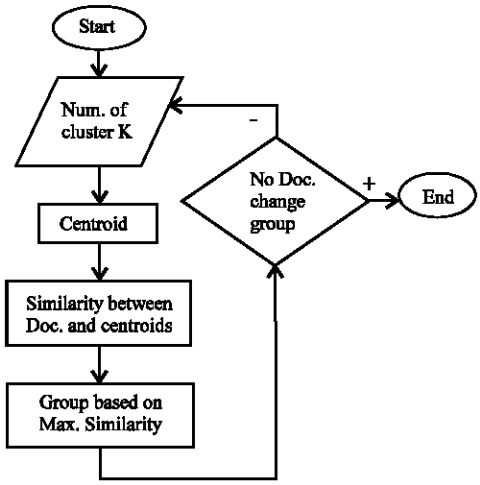

Fig. 1: K-means algorithm for text document clustering

Concepts of neighbor and link: To calculate the neighbor of two documents, we say the two documents are neighbors if their similarity is more than a threshold limit like this (Guha *et al.*, 1999):

$$\text{sim}(d_i, d_j) \geq \theta, 0 \leq \theta \leq 1 \tag{4}$$

In which the threshold value O is a parameter which is defined by the user. Then we use a neighbor matrix for a set of data containing n document which is a neighbor matrix n×n every input M (i,j) is either one or zero depending on the point that whether d_i and d_j are neighbors or not. The value of link function link(d_i,d_j) is defined as number of joint neighbor between d_i and d_j based on the following relation (Guha *et al.*, 1999):

$$\text{link}(d_i, d_j) = \sum_{m=1}^{n} M[i,m] * M[m,j] \tag{5}$$

K-means algorithm for text clustering: Figure 1 shows operation of K-means algorithm on text document clustering briefly.

Semantic similarity between documents based on ontology

Semantic vector space model: Disuse of semantic relations among words in presenting text data is the main difficulty of vector space model based on word. In this model, the importance of each word is weighted based on its existence in the documents and words behave independently, however they are semantically related. To overcome the weakness of the above model and for better organization of documents, we used semantic vector space model.

Conceptual features were introduced in (Bloehdom and Hotho, 2004; Hotho *et al.*, 2003) before. It is a collection of a few words which describes the same high level concept.

Three methods are introduced (Bloehdom and Hotho, 2004; Hotho *et al.*, 2003) to apply the conceptual features in the vector model as below:

- Adding conceptual features to the term space (term + concepts)
- Substituting the related terms with conceptual features
- Reducing the vector space model dimension by only conceptual features instead of terms

Then a phrase based model is used by Jurisica *et al.* (2004). It includes extracting the phrases of the documents and calculating the similarity between every couple of phrases. The both methods (based on the term and based on the phrase) change the main dimension of vector space model. The results of the previous researches show that only the (term + concepts) method between the term based models improves the performance rather than the base model but this method has some problems such as redundancy of the information and the dimensions. Besides that the problem of the phrase based model is extracting meaningful terms of the document which is a difficult process practically.

The model used in this article doesn't change the main dimensions of the vector space model (Jing *et al.*, 2010). In contrast, it uses semantic knowledge effectively in weighing words again in the vector space based on the term. So the new weight of each word in the ith document is calculated as follows:

$$\tilde{x}_{ij_1} = x_{ij_1} + \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{m} \delta_{j_1 j_2} x_{ij_2} \tag{6}$$

where, χ_{ij1} is the main weight of the word t_{j1} and χ_{ij2} is the main weight of all word t_{j2} in the vector space model which are related to t_{j1} and δ_{j1j2} is the similarity between the word t_{j1} and t_{j2}.

This relationship shows that if a word is related to other words a lot, its weight increases considerably. So based on this information, the dependence among words is created in semantic vector space model. Now we describe how to calculate δ.

Calculation of dependence of two words: We calculated the semantic similarity between two words using WordNet ontology which is an internet semantic dictionary (Miller, 1995). We need the similarity among concepts related to each word to obtain the similarity among words. The semantic similarity between c1 and c2 can be calculated through tree hierarchical structure in WordNet which there are different attitudes about it. Our proposed method in this article is a combination of the weight of the position of the concepts (Zhao, 1996) and the weight of the distance among concepts (Kolodner, 1993) which you can see in Eq. 7:

$$\text{sim}(c1,c2) = \frac{\text{cd}(c1,c2)}{\text{l}(c1,c2)} \times \frac{1}{\max[(\text{d}(c1) - \text{cd}(c1,c2)), (\text{d}(c2) - \text{cd}(c1,c2))]} \tag{7}$$

where, cd(c1,c2) equals the depth of common parent of c1 and c2 and l(c1,c2) is the length of the distance between the two concepts in ontology tree. Obviously, the more the depth of the common parent of the two concepts and the less the distance between them, the more the similarity. The denominator of the fraction is for making distinctions between the states with equal distance but with different positions relating to their common parent. Look at Fig. 2 for better explaining this relation. we obtain similarity of three couple (F,O) (K,N) and (M,N) according to the formula (7).

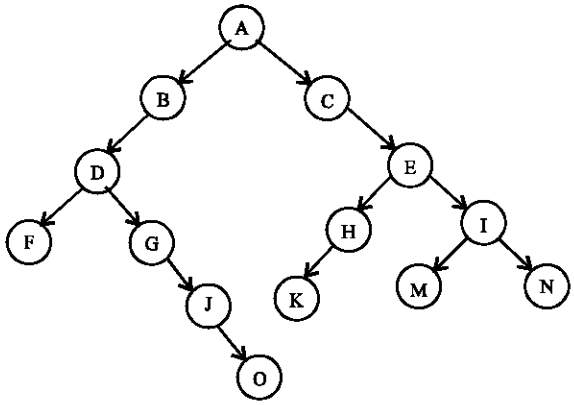

Fig. 2: A sample ontology

$$\text{sim}(F,O) = \frac{2}{4} \times \frac{1}{\max(1,3)} = 0.166$$

$$\text{sim}(K,N) = \frac{2}{4} \times \frac{1}{\max(2,2)} = 0.25$$

$$\text{sim}(M,N) = \frac{3}{2} \times \frac{1}{\max(1,1)} = 1.5$$

As it is clear in this tree, couples (F, O) and (K, N) have the same common parent depth and the same length path. But they are different as the amount of the similarity. Therefore to make difference between these models, we valuate the denominator of the fraction by the maximum length of the node to the common parent.

The couple nodes (M, N) are more similar rather than the other couple nodes because they are in the lower level of the ontology tree and therefore they have more special values. The diversity between the different models are shown in the concluded results clearly.

Finally we make them normal to use the obtained similarity value better in this way:

$$\text{NormSim}(c1,c2) = \frac{\text{sim}(c1,c2) - \text{MinSim}}{\text{MaxSim} - \text{MinSim}} \tag{8}$$

Calculating the similarity among words is the next step. The semantic similarity among words is presented using the maximum similarity of the concepts related to those words . If we suppose that words w_1 and w_2 contain a and b concepts, respectively the relationship (9) can show the semantic similarity between the two words:

$$\text{sim}(w_1, w_2) = \max\{\text{NormSim}(c_1, c_2)\} \tag{9}$$

$$c_1 \in \{c_{1,1}, c_{1,2}, \ldots, c_{1,a}\} \text{ and } c_2 \in \{c_{2,1}, c_{2,2}, \ldots, c_{2,b}\}$$

And if the two documents d_1 and d_2 have n and m words of WordNet, their semantic similarity is defined as follows:

$$sim_{WN}(d_1,d_2)=(\sum_{i=1}^{m}\sum_{j=1}^{n}sim(w_{1,i},w_{2,j}))/mn \quad (10)$$

Using semantic neighbors in clustering with k-means algorithm

Semantic neighbors: As in the semantic vector space model, we took into consideration the semantic relationships among words in the vector space, so this similarity among documents is not just based on their common words but contains their common concepts despite the existence of different words. Therefore the obtained neighbor is more accurate than the previous methods. The results obtained from applying semantic vector space model in calculating inter-document neighbor, called as semantic neighbor, in order to use them in k-means clustering algorithm, shows that it has a better performance than the previous methods.

Use of neighbor and link in k-means clustering: In this section we explain how to use suggested method for applying in the k-means clustering algorithm steps in short.

Determining the centers of initial clusters: As the efficiency of k-means algorithms are very sensitive to the selection of initial centers, their suitable distribution will be very effective in the obtained results.

We suppose that a cluster's documents are more similar to each other rather than different clusters' documents in the clustering. Therefore a good choice of a point to be the cluster's center, should be close enough to a special group of the documents. More ever, it should be separated from other centers truly. To evaluate the number of the documents which are more similar to the current document, we use the neighbors' numbers of a document in a data set with adjusting a proper similarity threshold θ.

As both cosine and link functions can calculate the amount of the similarity of two documents, we apply both of them to evaluate the dissimilarity of two documents which are the candidates of the initial centers.

So selecting initial centers is done based on three values: pairwise similarity calculated by cosine function, link function value and number of the neighbors of the documents in the set of data. This combination leads to a high quality selection of initial centers (Luo *et al*., 2009).

Determining similarity criterion: In order to determine the nearest cluster center for each document during the step of cluster identification, a criterion of assessment is needed. This similarity criterion is a combination of the functions of cosine and link. If we add the link criterion, the pairwise similarity of documents is improved and the documents available in the neighbor can improve the precision between the document and the cluster center. This relation is shown in formula (11):

$$f(d_i,c_j)=\alpha\times\frac{link(d_i,c_j)}{L_{max}}+(1-\alpha)\times\cos(d_i,c_j) \quad (11)$$

where, L_{max} is the largest value of $link(d_i,c_j)$ and α, that is between 0 and 1, is the coefficient set by the user (Luo *et al*., 2009).

EXPERIMENTS AND RESULTS

We used a set of real-world text data to show the improved performance of our proposed method and compared the results of the obtained clustering with the basic k-means algorithm and previous

Table 1: Features of data sets

Data set	NO. doc	NO. class	Min. class size	Max. class size	NO. term
TOP1	447	5	19	193	2573
TOP2	839	7	9	433	3911
MED1	92	3	27	37	456
MED2	240	10	11	39	1058
CISI1	203	4	8	126	782
CISI2	154	4	13	82	677
CISI3	109	3	10	70	455

methods based on neighbors. We also used Visual C# 64-bit operating system of windows vista with a 6.00 GB memory and a 2.40 GHz processor.

Data sets: We used 7 sets of data obtained from two different kinds of text databases. The first two sets were chosen from the database of Reuter-21578 Distribution 1.0 (Lewis, 2004) with the subject of TOPIC. The next 5 sets were extracted from a Classic text database which the first two sets related to MEDLINE and the next 3 sets related to the dataset CISI. Their specifications are summarized in Table 1.

In addition we used a number of steps for preprocessing data which are as follows: conversion of documents into a set of words, omission of stop words and finding the origin of the words using porter stemmer (Porter, 1980). Moreover, any word stem that occurred fewer than two documents in one collection was eliminated. Finally, we use word stems as features or dimensions of the document vector space model.

Clustering results: We used F-measure criterion for the assessment of the accuracy of the proposed algorithm and for its comparison with the previous methods. The F-measure is a harmonic combination of the precision and recall values used in information retrieval (Van Rijsbergen, 1979). Precision of cluster C_i is defined to be the number of texts correctly assigned divided by the total number of texts in cluster C_i. We define Recall of cluster C_i to be the number of texts correctly assigned divided by the total number of texts that should be assigned. Let P be average precision. Let R be average recall. F-measure is defined to be 2 RP/(R+P).

The obtained results are shown in Fig. 3 in which we had an average of 10 runs for each method and each dataset.

The first column of the diagram (KM) is the results of basic k-means algorithm and the next three columns are related to the previous studies in the field of using neighbor and link in clustering through k-means algorithm. the first (NC), second (NS) and third (NCS) columns are related to the use of neighbor in determining initial centers, similarity criterion and determining initial centers and similarity criterion in combination, respectively.

The last three columns show the results obtained from our proposed method. The fifth column (SNC) is the result of using semantic neighbors in determining initial centers, the sixth column (SNS) the result of using semantic neighbors in similarity criterion and the last column (SNCS) is the result of using semantic neighbors in the two methods mentioned in combination.

As you can see in the diagram, in all cases, applying the concept of semantic neighbor in determining initial centers and also in its combination with similarity criterion, a considerable improvement was observed compared with the all mentioned methods. Using semantic neighbor alone in determining similarity criterion is comparable with the best previous work in most of cases. And it is more accurate than other previous methods.

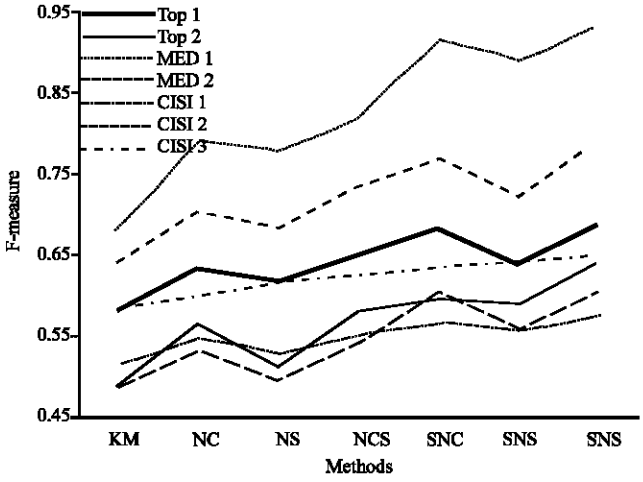

Fig. 3: Comparison of the result of suggested method with basic K-means algorithm and previous methods

CONCLUSION

In this study, we presented semantic neighbor method to improve the performance of k-means algorithms in which we used the concept of semantic similarity in reweighing word frequency in the vector space model of documents. The results obtained from our proposed method in the set of real-world text data shows a considerable increase in the precision of clustering compared with the basic clustering algorithm and previous methods. In future studies, we plan to investigate the concept of neighbor more detailed and present a method to make it more accurate in order to improve the clustering results as much as possible.

REFERENCES

Bloehdom, S. and A. Hotho, 2004. Text classification by boosting weak learners based on terms and concepts. Proceeding of the 4th IEEE International Conference on Data Mining, November 1-4, 2004, Brighton, UK., pp: 331-334.

Guha, S., R. Rastogi and K. Shim, 1999. ROCK: A robust clustering algorithm for categorical attributes. Inform. Syst., 25: 345-366.

Hirst, G. and D. St-Onge, 1997. Lexical Chains as Representation of Context for the Detection and Correction Malapropisms. In: WordNet: An Electronic Lexical Database, Fellbaum, C. (Ed.). The MIT Press, Cambridge, MA, pp: 305-332.

Hotho, A., S. Staab and G. Stumme, 2003. Wordnet improves text document clustering. Proceedings of the SIGIR 2003 Semantic Web Workshop, (SWW'03), Toronto, Canada, pp: 541-544.

Jain, A.K. and R.C. Dubes, 1998. Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs, New Jersey, ISBN: 013022278X.

Jing, L., M.K. Ng and J.Z. Huang, 2010. Knowledge-based vector space model for text clustering. Knowledge Inform. Syst., 25: 35-55.

Jurisica, I., J. Mylopolous and E. Yu, 2004. Ontologies for knowledge management: An information systems perspective. Knowledge Inform. Syst., 6: 380-401.

Kolodner, J., 1993. Case-Based Reasoning. Morgan Kaufmann Publishers, San Mateo, CA., New York.

Lewis, D.D., 2004. Reuters-21578 text categorization test collection distribution 1.0. http://www.daviddlewis.com/resources/testcollections/reuters21578/readme.txt

Li, Y. and C. Luo, 2008. Text clustering with feature selection by using statistical data. IEEE Transact. Knowledge Data Eng., 20: 641-652.

Luo, C., Y. Li and S.M. Chung, 2009. Text document clustering based on neighbors. Data Knowledge Eng., 68: 1271-1288.

Mao, W. and W.W. Chu, 2002. Free text medical document retrieval via phrased-based vector space model. Proceeding of the Annual Symposium of the American Medical Informatics Association, November 9-13, 2002, San Antonio, TX., USA., pp: 489-493.

Miller, G.A., 1995. Word net: A lexical database for English. Commun. ACM, 38: 39-41.

Porter, M.F., 1980. An algorithm for suffix stripping. Program, 14: 130-137.

Steinbach, M., G. Karypis and V. Kumar, 2000. A comparison of document clustering techniques. Proceedings of the 6th ACM SIGKDD World Text Mining Conference (TMW'2000), Boston, MA., pp: 1-2.

Van Rijsbergen, C.J., 1979. Information Retrieval. 2nd Edn., Buttersworth, London, UK., Pages: 224.

Zhao, G., 1996. Analogical translator: Experience-guided transfer in machine translation. Ph.D. Thesis, University of UMIST, UK.