Volume 3, Issue 8, August 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Document Clustering on Various Similarity Measures

Ms.K.Sruthi
Pursuing M.Tech(CSE)
Anurag Group of Institutions, India

Mr.B.Venkateshwar Reddy
Asst.Professor
Anurag Group of Institutions, India

Abstract: Clustering is a useful technique that organizes a large quantity of unordered text documents into a small number of meaningful and coherent clusters, thereby providing a basis for intuitive and informative navigation and browsing mechanisms. A wide variety of distance functions and similarity measures have been used for clustering. In this paper we mainly focuses on different similarity measures, view points and Document clustering. We introduce a novel multi-viewpoint based similarity measure and two related clustering methods. Using multiple viewpoints, more informative assessment of similarity could be achieved.

Key Terms—Document clustering, similarity measures,Criterion functions.

I. INTRODUCTION

Document clustering groups similar documents that to form a coherent cluster, while documents that are different have separated apart into different clusters. Clustering of text documents plays a vital role in efficient Document Organization, Summarization, Topic Extraction and Information Retrieval. Initially used for improving the precision or recall in an Information Retrieval System .more recently, clustering has been proposed for use in browsing a collection of documents or in organizing the results returned by a search engine in response to user's query or help users quickly identify and focus on the relevant set of results. However, the definition of a pair of documents being similar or different is not always clear and normally varies with the actual problem setting. For example, when clustering research papers, two documents are regarded as similar if they share similar thematic topics. When clustering is employed on web sites, we are usually more interested in clustering the component pages according to the type of information that is presented in the page. For instance, when dealing with universities' web sites, we may want to separate professors' home pages from students' home pages, and pages for courses from pages for research projects. This kind of clustering can benefit further analysis and utilize of the dataset such as information retrieval and information extraction, by grouping similar types of information sources together. Accurate clustering requires a precise definition of the closeness between a pair of objects, in terms of either the pair wised similarity or distance. A variety of similarity or distance measures have been proposed and widely applied, such as Euclidean distance, cosine similarity, Jaccard coefficient, Pearsoncorrelation coefficient. Given the diversity of similarity and distance measures available, their effectiveness in text document clustering is still not clear.The traditional dissimilarity or similarity measure and ours is that the former uses only a single viewpoint, which is the origin by using a specific measure . By using this measure less informative assessment of similarity could be achieved. We propose a Multiviewpoint-based Similarity measuring method, named MVS. MVS is potentially more suitable for text documents than the popular cosine similarity. The key contribution of this paper is the fundamental concept of similarity measure from multiple viewpoints. Multiview point similarity measure for document clustering which provides maximum efficiency and performance. Scope of this mvs is measure similarty and dissimilarity between objects which are present in different clusters. Two criterion functions for document clustering are proposed based on this new measure. which are fast and scalable like k-means, but are also capable of providing high-quality and consistent performance.It provides best and more accuracy in results .

II. DOCUMENT CLUSTERING:

Given a set *S* of *n* documents, we would like to partition them into a pre-determined number of *k* subsets S1, S2, . . . , Sk , such that the documents assigned to each subset are more similar to each other than the documents assigned to different subsets. Document clustering techniques mostly rely on single term analysis of the document data set, such as the Vector Space Model. To achieve more accurate document clustering, more informative features including phrases and their weights are particularly important in such scenarios.

- Document clustering is particularly useful in many applications such as automatic categorization of documents, grouping search engine results, building taxonomy of documents, and others.
- Each document in a corpus corresponds to an m-dimensional vector d, where 'm' is the total number of terms.
- Document vectors are often subjected to some weighting schemes, such as the standard Term Frequency-Inverse Document Frequency (TF-IDF), and normalized to have unit length.

A.Document pre-processing steps

- Tokenization:

A document is treated as a string (or bag of words), and then partitioned into a list of tokens.

- Removing stop words:
  Stop words are frequently occurring, insignificant words. This step eliminates the stop words.
- Stemming word:
  This step is the process of conflating tokens to their root form.

B.Document representation

Generating N-distinct words from the corpora and call them as index terms (or the vocabulary). The document collection is then represented as a N-dimensional vector in term space.

Computing Term weights

Term Frequency.

Inverse Document Frequency.

Compute the TF-IDF weighting.

C.TFIDF Analysis

By taking into account these two factors : term frequency (TF) and inverse document frequency (IDF) it is possible to assign weights to search results and therefore ordering them statistically. Put another way a search result's score Ranking is the product of TF and IDF: **TFIDF = TF * IDF**

where:

- ∗ TF = C / T where C = number of times a given word appears in a document and T = total number of words in a document.
- ∗ Document IDF = D / DF where D = total number of documents in a corpus, and DF = total number of documents containing a given word.

III. SIMILARITY MEASURES:

Before clustering, a similarity/distance measure must be determined. The measure reflects the degree of closeness or separation of the target objects and should correspond to the characteristics that are believed to distinguish the clusters embedded in the data. All clustering methods have to assume some cluster relationship among the data objects that they are applied on. Similarity between a pair of objects can be defined either explicitly or implicitly. The nature of similarity measure plays a very important role in the success or failure of a clustering method. some of the similarity measures explained briefly below based on single view point and multiviewpoint

A. Euclidean Distance:

Euclidean distance is a regular metric for geometrical problems. It is the common distance between two points and can be without difficulty measured with a ruler in two- or threedimensional space. It is also the default distance measure used with the K-means algorithm. Euclidean distance is one of the most popular measures:

Dist(di, dj) = | |di − dj ||. It is used in the traditional k-means algorithm. The objective of k-means is to minimize the Euclidean distance between objects of a cluster and that cluster's centroid. Measuring distance between text documents, given two documents da and db represented by their term vectors

ta and tb respectively, the Euclidean distance of the two documents is defined as

$$D_E(\vec{t_a}, \vec{t_b}) = \left(\sum_{t=1}^{m} |w_{t,a} - w_{t,b}|^2\right)^{1/2},$$

where the term set is T = {t1, . . . , tm}. we use the tfidf value as term weights.

B.Cosine Similarity:

Cosine similarity is one of the most popular similarity measure practical to text documents, such as in various information retrieval applications and clustering too. An important property of the cosine similarity is its independence of document length. For two documents di and dj, the similarity between them can be calculated

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

Since the document vectors are of unit length,the above equation is simplified to:

$$cos\,(d_i, d_j) = d_i \,.\, d_j$$

When the cosine value is 1 the two documents are identical, and 0 if there is nothing in common between them (i.e., their document vectors are orthogonal to each other).

C. Jaccard Coefficient:

The Jaccard coefficient, which is sometimes referred to as the Tanimoto coefficient, measures similarity as the intersection divided by the union of the objects. For text document, the Jaccard coefficient compares the sum weight of shared terms to the sum weight of terms that are present in either of the two document but are not the shared terms.

$$\mathrm{Sim}_{eJacc}\left[u_i, u_j\right] = \frac{u_i^t u_j}{||u_i||^2 + ||u_j||^2 - u_i^t u_j}$$

The Jaccard coefficient is a similarity measure and ranges

between 0 and 1. It is 1 when the Ui=Uj and 0 whenUi and Uj are disjoint, where 1 means the two objects are the same and 0 means they are completely different.

D.Pearson Correlation Measure:

Correlation clustering provides a method for clustering a set of objects into the best possible number of clusters, without specifying that number in proceed. Correlation Clustering that does not require a bound on the number of clusters that the data is partitioned into. Rather, Correlation Clustering divides the data into the optimal number of clusters based on the similarity between the data points. Given the term set T = {t1, ... tm}, a commonly used form is

$$SIM_P(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{m\sum_{t=1}^{m} w_{t,a} \times w_{t,b} - TF_a \times TF_b}{\sqrt{[m\sum_{t=1}^{m} w_{t,a}^2 - TF_a^2][m\sum_{t=1}^{m} w_{t,b}^2 - TF_b^2]}}$$

where $TF_a = \sum_{t=1}^{m} w_{t,a}$ and $TF_b = \sum_{t=1}^{m} w_{t,b}$.

However, unlike the other measures, it ranges from +1 to −1 and it is 1 when $\overrightarrow{t_a} = \overrightarrow{t_b}$.

The Euclidean distance is a distance measure, while the cosine similarity, Jaccard coefficient and Pearson coefficient are similarity measures. We apply a simple transformation to convert the similarity measure to distance values. Because both cosine similarity and Jaccard coefficient are bounded in [0, 1]and monotonic, we take D = 1 − SIM as the corresponding distance value. For Pearson coefficient, which ranges from −1 to +1, we take D = 1 − SIM when SIM ≥ 0 and D = |SIM| when SIM < 0.

E. Multiviewpoint-Based Similarity Measure:

Using multiple viewpoints, more informative assessment of similarity could be achieved. Two criterion functions for document clustering are proposed based on this new measure. We compare them with several well-known clustering algorithms that use other popular similarity measures on various document collections to verify the advantages of our proposal

The cosine similarity can be expressed in the following form without changing its meaning: **Sim(di, dj) = cos(di−0, dj−0) = (di−0)t (dj−0)** where 0 is vector 0 that represents the origin point. According to this formula, the measure takes 0 as one and only reference point. The similarity between two documents di and dj is determined w.r.t. the angle between the two points when looking from the origin. similarity of two documents di and dj given that they are in the same cluster is defined as the average of similarities measured relatively from the views of all other documents outside that cluster. What is interesting is that the similarity here is defined in a close relation to the clustering problem. The two objects to be measured must be in the same cluster, while the points from where to establish this measurement must be outside of the cluster. We call this proposal the Multiviewpoint-based Similarity, or MVS.

The similarity between two points di and dj inside cluster Sr, viewed from a point dh outside this cluster, is equal to the product of the cosine of the angle between di and dj looking from dh and the euclidean distances from dh to these two points.

$$\begin{aligned}
&\mathrm{MVS}(d_i, d_j | d_i, d_j \in S_r) \\
&\quad = \frac{1}{n - n_r} \sum_{d_h \in S \setminus S_r} (d_i - d_h)^t (d_j - d_h) \\
&\quad = \frac{1}{n - n_r} \sum_{d_h} \cos(d_i - d_h, d_j - d_h) \|d_i - d_h\| \|d_j - d_h\|.
\end{aligned}$$

Algorithm: **MVS similarity Matrix**

```
procedure BUILDMVSMATRIX($A$)
    for $r \leftarrow 1 : c$ do
        $D_{S\setminus S_r} \leftarrow \sum_{d_i \notin S_r} d_i$
        $n_{S\setminus S_r} \leftarrow |S \setminus S_r|$
    end for
    for $i \leftarrow 1 : n$ do
        $r \leftarrow$ class of $d_i$
        for $j \leftarrow 1 : n$ do
            if $d_j \in S_r$ then
                $a_{ij} \leftarrow d_i^t d_j - d_i^t \frac{D_{S\setminus S_r}}{n_{S\setminus S_r}} - d_j^t \frac{D_{S\setminus S_r}}{n_{S\setminus S_r}} + 1$
            else
                $a_{ij} \leftarrow d_i^t d_j - d_i^t \frac{D_{S\setminus S_r} - d_j}{n_{S\setminus S_r} - 1} - d_j^t \frac{D_{S\setminus S_r} - d_j}{n_{S\setminus S_r} - 1} + 1$
            end if
        end for
    end for
    return $A = \{a_{ij}\}_{n \times n}$
end procedure
```

IV. CLUSTERING CRITERION FUNCTIONS:

A. ***Internal Criterion Functions*:** This class of clustering criterion functions focuses on producing a clustering solution that optimizes a particular criterion function that is defined over the documents that are part of each cluster and does not take into account the documents assigned to different clusters. Due to this intra-cluster view of the clustering process we will refer to these criterion functions as **internal**. The first internal criterion function that we will study maximizes the sum of the average pairwise similarities between the documents assigned to each cluster, weighted according to the size of each cluster .Criterion function is similar to that used in the context of hierarchical agglomerative clustering that uses the group-average heuristic to determine which pair of clusters to merge next. The second criterion function that we will study is used by the popular vector-space variant of the *K*-means algorithm In this algorithm each cluster is represented by its centroid vector and the goal is to find the clustering solution that maximizes the similarity between each document and the centroid of the cluster that is assigned to the last internal criterion function that we will study is that used by the traditional *K*-means algorithm. This criterion function uses the Euclidean distance to determine which documents should be clustered together, and determines the overall quality of the clustering solution by using the sum-of-squared-errors function.

B. External Criterion Functions

Unlike internal criterion functions, external criterion functions derive the clustering solution by focusing on optimizing a function that is based on how the various clusters are different from each other. Due to this inter-cluster view of the clustering process we will refer to these criterion functions as external.

V. CRITERION FUNCTION OPTIMIZATION

common way of performing criterion function optimization is to use a greedy strategy Our greedy optimizer consists of two phases: (i) **initial clustering**, and (ii) **cluster refinement**.

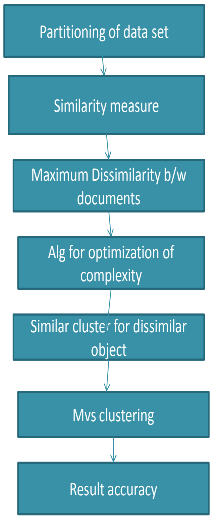
Partitioning of data set
Similarity measure
Maximum Dissimilarity b/w documents
Alg for optimization of complexity
Similar cluster for dissimilar object
Mvs clustering
Result accuracy
5

fig.dataflow diagram

In the initial clustering phase, a clustering solution is computed as follows. If *k* is the number of desired clusters, *k* documents are randomly selected to form the seeds of these clusters. The similarity of each document to each of these *k* seeds is computed, and each document is assigned to the cluster corresponding to its most similar seed. The similarity between documents and seeds is determined using the cosine measure of the corresponding document vectors.

The goal of the cluster refinement phase is to take the initial clustering solution and iteratively refine it. During each iteration, the n documents are visited one by one in a totally random order. Each document is checked if its move to another cluster results in improvement of the objective function. If yes, the document is moved to the cluster that leads to the highest improvement. If no clusters are better than the current cluster, the document is not moved. The clustering process terminates when an iteration completes without any documents being moved to new clusters. Unlike the traditional k-means, this algorithm is a stepwise optimal procedure. While k-means only updates after all n documents have been reassigned, the incremental clustering algorithm updates immediately whenever each document is moved to new cluster.

VI. Conclusion

To conclude, this investigation found that except for the Euclidean distance measure, the other measures have comparable effectiveness for the partitional text document clustering task. The Jaccard and Pearson coefficient measures find more coherent clusters. Despite of the above differences, these measures' overall performance is similar. So we introduced multi-viewpoint based similarity measure and two related clustering methods.Using multiple viewpoints, more informative assessment of similarity could be achieved and performance is much betterthan above similarity measures.

Future methods could make use of the same principle, but define alternative forms for the relative similarity or do not use average but have other methods to combine the relative similarities according to the different viewpoints. Besides, this paper focuses on partitional clustering of documents. In the future, it would also be possible to apply the proposed criterion functions for hierarchical clustering algorithms. Finally, we have shown the application of MVS and its clustering algorithms for text data. It would be interesting to explore how they work on other types of sparse and high-dimensional data.

References:

[1] S. Zhong, "Efficient Online Spherical K-means Clustering," Proc.IEEE Int'l Joint Conf. Neural Networks (IJCNN), pp. 3180-3185, 2005.

[2] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh, "Clustering withBregman Divergences," Mach ine Learning Research pp. 1705-1749, Oct. 2005.

[3] M. Pelill,What Is a Cluster? PerspectiveGame the ory, "Proc. NIPS Worksho Clustering Theory, 2009.

[4] D. Lee and J. Lee "Dynamic Dissimilarity measure for Support Based Clustering, IEEE tran .knowledge and Data Eng., vol. 2 no. 6, pp. 900-905, June .

[5] A. Banerjee, I. Dhillon, J. Ghosh, and S Sra, clustering on the Unit Hypersphere UsingVonmisesfisher Distributions,"J. Machin Learning Research, vol. 6 Sept. 2005.

[6] W. Xu, X. Liu, and Y. Gong, "Document Clustering Based on Non-Negative Matrix Factorization, Proc. 26th Ann.Int'l ACM SIGIRConf. Research and dev elopment in Informaion Retrieval, pp. 267-273 2003.

[7] I.S. Dhillon, S. Mallela, and D.S .Modha informatiion-Theoretic Co-Clustering, Proc 9th ACM SIGKDD int Conf. Knowledge Discovery and Data Mining (KDD), pp. 89-98, 2003.

[8] S. Zhong and J. Ghosh, "A Comparative Study of Generative Models forDocument Clustering," Proc. SIAM Int'l Conf. Data Mining WorkshopClustering High dimen-sional Data and Its Applications, 2003.

[9] Y. Zhao and G. Karypis, "Criterion Functions for Document Clustering: Experiments and Analysis," technical report, Dept. of computer Science, Univ. of Minnesota, 2002.

[10] E.-H. Han, D. Boley, M. Gini, R. Gross, KHas tings, G. Karypis, V.Kumar, B. Mobasher, and J. Moore, "Webace: A Web Agent for Document Categorization and Exploration," Proc. Second Int'lConf. Autonomous Agents (AGENTS '98), pp. 408-415, 1998.

[11] A. Strehl, J. Ghosh, and R. Mooney,Impact of si Milarity Measures on Web-Page Clustering," Proc. 17th Nat'l Conf. ArtificiaIntelligence: Workshop of ArtificialIntelligence for Web search (AAAI),pp. 58-64, July 2000.

AUTHORS BIOGRAPHY

Ms.K.Sruthi received B.Tech Computer Science andEngineering from JNTUH.Pursuing M.Tech Computer Science and Engineering from Anurag group of Institutions Hyderabad,India. Her area of interest includes Data Mining, Machine Learning and Pattern Recognition.

Mr.B.Venkateshwar Reddy Received M.Sc Mathematics from Osmania University and M.E Computer Science and Engineering from Sathyabama University,
Chennai. Presently working as a Assistant Professor in school of Engineering, Anurag Group of Institutions, Hyderabad, India. Published three papers in various National and International Conferences, Journals.