International Journal of Engineering Inventions
ISSN: 2278-7461, www.ijeijournal.com
Volume 1, Issue 3 (September 2012) PP: 01-05

Clustering With Multi-Viewpoint Based Similarity Measure: An Overview

Mrs. Pallavi J. Chaudhari[1], Prof. Dipa D. Dharmadhikari[2]
[1]*Lecturer in Computer Technology Department, MIT College of Polytechnic, Aurangabad*
[2]*Assistant Professor in Computer Sci. and Engineering Department, MIT College of Engineering, Aurangabad*

Abstract—Clustering is a useful technique that organizes a large quantity of unordered text documents into a small number of meaningful and coherent cluster, thereby providing a basis for intuitive and informative navigation and browsing mechanisms. There are some clustering methods which 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 major difference between a traditional dissimilarity/similarity measure and ours is that the former uses only a only a single viewpoint, which is the origin, while the latter utilizes many different viewpoints, which are objects assumed to not be in the same cluster with the two objects being measured. Using multiple viewpoints, more informative assessment of similarity could be achieved. Theoretical analysis and empirical study are conducted to support this claim. 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.

Keywords—Document Clustering, Multi-Viewpoint Similarity Measure, Text Mining.

I. INTRODUCTION

Clustering in general is an important and useful technique that automatically organizes a collection with a substantial number of data objects into a much smaller number of coherent groups [1] .The aim of clustering is to find intrinsic structures in data, and organize them into meaningful subgroups for further study and analysis. There have been many clustering algorithms published every year. They can be proposed for very distinct research fields, and developed using totally different techniques and approaches. Nevertheless, according to a recent study [2] more than half a century after it was introduced; the simple algorithm k-means still remains as one of the top 10 data mining algorithms nowadays. It is the most frequently used partitional clustering algorithm in practice. Another recent scientific discussion [3] states that *k*-means is the favorite algorithm that practitioners in the related fields choose to use. *K*-means has more than a few basic drawbacks, such as sensitiveness to initialization and to cluster size, difficulty in comparing quality of the clusters produced and its performance can be worse than other state-of-the-art algorithms in many domains. In spite of that, its simplicity, understandability and scalability are the reasons for its tremendous popularity. While offering reasonable results, *k*-means is fast and easy to combine with other methods in larger systems. A common approach to the clustering problem is to treat it as an optimization process. An optimal partition is found by optimizing a particular function of similarity (or distance) among data. Basically, there is an implicit assumption that the true intrinsic structure of data could be correctly described by the similarity formula defined and embedded in the clustering criterion function. Hence, effectiveness of clustering algorithms under this approach depends on the appropriateness of the similarity measure to the data at hand. For instance, the original *K-means* has sum-of-squared-error objective function that uses Euclidean distance. In a very sparse and high dimensional domain like text documents, spherical *k*-means, which uses cosine similarity instead of Euclidean distance as the measure, is deemed to be more suitable [4],[5]. A variety of similarity or distance measures have been proposed and widely applied, such as cosine similarity and the Jaccard correlation coefficient. Meanwhile, similarity is often conceived in terms of dissimilarity or distance [6].Measures such as Euclidean distance and relative entropy has been applied in clustering to calculate the pair-wise distances.

The Vector-Space Model is a popular model in the information retrieval domain [7] .In this model, each element in the domain is taken to be a dimension in a vector space. A collection is represented by a vector, with components along exactly those dimensions corresponding to the elements in the collection. One advantage of this model is that we can now weight the components of the vectors, by using schemes such as TF-IDF [8].The Cosine-Similarity Measure *(CSM*) defines similarity of two document vectors d_i and d_j , $sim(d_i , d_j)$, as the cosine of the angle between them. For unit vectors, this equals to their inner product:

$$\sin(d_i, d_j) \ = \ \cos \ (d_i, d_j) \ = \ d_i^{\,t} d_j \qquad (1)$$

This measure has proven to be very popular for query-document and document-document similarity in text retrieval. Collaborative-filtering systems such as GroupLens [9] use a similar vector model, with each dimension being a “vote” of the user for a particular item. However, they use the Pearson Correlation Coefficient as a similarity measure, which first subtracts the average of the elements from each of the vectors before computing their cosine similarity. Formally, this similarity is given by the formula:

$$c\ (X,Y)=\frac{\sum_j (x_j-\bar{x})(y_j-\bar{y})}{\sqrt{\sum_j (x_j-\bar{x})^2 \sum_j (y_j-\bar{y})^2}} \tag{2}$$

Where, x_j is the value of vector X in dimension j , x is the average value of X along a dimension, and the summation is over all dimensions in which both X and Y are nonzero [9]. Inverse User Frequency may be used to weight the different components of the vectors. There have also been other enhancements such as default voting and case amplification [10], which modify the values of the vectors along the various dimensions. In a provocative study, Ahlgren et al. questioned the use of Pearson's Correlation Coefficient as a similarity measure in Author Co-citation Analysis (ACA) with the argument that this measure is sensitive for zeros. Analytically, the addition of zeros to two variables should add to their similarity, but the authors show with empirical examples that this addition can depress the correlation coefficient between these variables. Salton's cosine is suggested as a possible alternative because this similarity measure is insensitive to the addition of zeros [7].In a reaction White defended the use of the Pearson correlation hitherto in ACA with the pragmatic argument that the differences between using different similarity measures can be neglected in the research practice. He illustrated this with dendrograms and mappings using Ahlgren et al.'s own data. Bensman contributed to the discussion with a letter in which he argued for using Pearson's r for additional reasons. Unlike the cosine, Pearson's r is embedded in multivariate statistics and because of the normalization implied this measure allows for negative values. The problem with the zeros can be solved by applying a logarithmic transformation to the data. In his opinion, this transformation is anyhow advisable in the case of a bivariate normal distribution. Leydesdorff & Zaal experimented with comparing results of using various similarity criteria—among which the cosine and the correlation coefficient—and different clustering algorithms for the mapping. Indeed, the differences between using the Pearson's r or the cosine were also minimal in our case. However, our study was mainly triggered by concern about the use of single linkage clustering in the ISI's World Atlas of Science [11]. The choice for this algorithm had been made by the ISI for technical reasons given the computational limitations of that time. The differences between using Pearson's Correlation Coefficient and Salton's cosine are marginal in practice because the correlation measure can also be considered as a cosine between normalized vectors [12]. The normalization is sensitive to the zeros, but as noted this can be repaired by the logarithmic transformation. More generally, however, it remains most worrisome that one has such a wealth of both similarity criteria (e.g., Euclidean distances, the Jaccard index, etc.) and clustering algorithms (e.g., single linkage, average linkage, Ward's mode, etc.) available that one is able to generate almost any representation from a set of data [13]. The problem of how to estimate the number of clusters, factors, groups, dimensions, etc. is a pervasive one in multivariate analysis. In cluster analysis and multi dimensional scaling, decisions based upon visual inspection of the results are common.

The following Table 1 summarizes the basic notations that will be used extensively throughout this paper to represent documents and related concepts.

TABLE 1
Notations

Notation	Description
n	number of documents
m	number of terms
c	number of classes
k	number of clusters
d	document vector, || d || = 1
S = { d1,….,dn}	set of all the documents
S_r	set of documents in cluster r
$D=\sum_{di\in S} d_i$	composite vector of all the documents
$D_r=\sum_{di\in Sr} d_i$	Composite vector of cluster r
C = D / n	centroid vector of all the documents
$C_r = D_r / n_r$	centroid vector of cluster r, $n_r = s_r$

II. RELATED WORK

2.1 Clustering:

Clustering can be considered the most unsupervised learning technique; so , as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. Clustering is the process of organizing objects into groups whose members are similar in some way.Therefore a cluster is a collection of objects which are similar between them and are dissimilar to the objects belonging to other clusters. Generally, clustering is used in Data Mining, Information Retrieval, Text Mining, Web Analysis, Marketing and Medical Diagnostic.

2.2 Document representation:.

The various clustering algorithms represents each document using the well-known term frequency-inverse document frequency *(tf-idf)* vector-space model (Salton, 1989). In this model, each document d is considered to be a vector in the term-space and is represented by the vector

$$d_{tfidf} = (tf_1 \log\ (n/df_1), tf_2 \log\ (n/df_2), ..., tf_m \log\ (n/df_m)) \qquad (3)$$

Where tf_i is the frequency of the i^{th} term (i.e., term frequency), *n* is the total number of documents, and df_i is the number of documents that contain the i^{th} term (i.e., document frequency). To account for documents of different lengths, the length of each document vector is normalized so that it is of unit length. In the rest of the paper, we will assume that the vector representation for each document has been weighted using *tf-idf* and normalized so that it is of unit length.

2.3 Similarity measures:

Two prominent ways have been proposed to compute the similarity between two documents d_i and d_j. The first method is based on the commonly-used (Salton, 1989) cosine function:

$$\cos\ (d_i, d_j)\ =\ d_i^t d_j\ /(\|d_i\|\ \ \|d_j\|) \qquad (4)$$

Since the document vectors are of unit length, it simplifies to $d_i^t d_j$. The second method computes the similarity between the documents using the Euclidean distance *dis* $(d_{i,}\ d_j) = \|d_i - d_j\|$. Note that besides the fact that one measures similarity and the other measures distance, these measures are quite similar to each other because the document vectors are of unit length.

III. OPTIMIZATION ALGORITHM

Our goal is to perform document clustering by optimizing criterion functions I_R and I_V *[clustering with MVSC]*. To achieve this, we utilize the sequential and incremental version of *k*-means [14] [15], which are guaranteed to converge to a local optimum. This algorithm consists of a number of iterations: initially, *k* seeds are selected randomly and each document is assigned to cluster of closest seed based on cosine similarity; in each of the subsequent iterations, the documents are picked in random order and, for each document, a move to a new cluster takes place if such move leads to an increase in the objective function. Particularly, considering that the expression of I_V *[clustering with MVSC]* depends only on n_r and $D_{r,}$ *r=1 … k,* let us represent I_V in a general form

$$I_v = \sum_{r=1}^{k} I_r\ (n_r, D_r) \qquad (5)$$

Assume that, at beginning of some iteration a document d_i belongs to a cluster S_P that has objective value I_P (n_P, D_P).d_i will be moved to another cluster S_q that has objective value I_q (n_q,D_q) if the following condition is satisfied:

$$\Delta I_v\ =\ I_p\ (n_p - 1, D_p - d_i)\ +\ (I_q\ (n_q + 1, D_q + d_i)\ -\ I_p(n_{p,} D_p)\ -\ I_q(n_{q,} D_q) \qquad (6)$$

$$st.q\ =\ \arg\max\ \{I_r\ (n_r + 1, D_r + d_i)\ -\ I_r\ (n_{r,} D_r)\}$$

Hence, document d_i is moved to a new cluster that gives the largest increase in the objective function, if such an increase exists. The composite vectors of corresponding old and new clusters are updated instantly after each move. If a maximum number of iterations is reached or no more move is detected, the procedure is stopped. A major advantage of our clustering functions under this optimization scheme is that they are very efficient computationally. During the optimization process, the main computational demand is from searching for optimum clusters to move individual documents to, and updating composite vectors as a result of such moves. If *T* denotes the number of iterations the algorithm takes, nz the total number of non-zero entries in all document vectors, the computational complexity required for clustering with I_R *and* I_V is approximately O (nz.k.T).

IV. EXISTING SYSTEM

The principle definition of clustering is to arrange data objects into separate clusters such that the intra-cluster similarity as well as the inter-cluster dissimilarity is maximized. The problem formulation itself implies that some forms of measurement are needed to determine such similarity or dissimilarity. There are many state-of-the art clustering approaches that do not employ any specific form of measurement, for instance, probabilistic model based method [16], and non-negative matrix factorization [17] .Instead of that Euclidean distance is one of the most popular measures. 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:

$$\min \sum_{r=1}^{k} \sum_{d_i \varepsilon S_r} \|\ d_i - C_r\ \|^2 \qquad (7)$$

However, for data in a sparse and high-dimensional space, such as that in document clustering, cosine similarity is more widely used. It is also a popular similarity score in text mining and information retrieval [18]. Cosine measure is used in a variant of *k*-means called spherical *k*-means [4]. While *k*-means aims to minimize Euclidean distance, spherical *k*-means intends to maximize the cosine similarity between documents in a cluster and that cluster's centroid:

$$\max\ \sum_{r=1}^{k} \sum_{d_i \varepsilon S_r} \frac{d_i^t C_r}{\|C_r\|} \qquad (8)$$

The major difference between Euclidean distance and cosine similarity, and therefore between k-means and spherical k-means, is that the former focuses on vector magnitudes, while the latter emphasizes on vector directions. Besides direct application in spherical k-means, cosine of document vectors is also widely used in many other document clustering methods as a core similarity measurement. The cosine similarity in Eq. (1) can be expressed in the Following form without changing its meaning:

$$\sin\ (d_i, d_j) = \cos\ (d_i - 0, d_j - 0) = (d_i - 0)^t\ (d_j - 0) \qquad (9)$$

Where, 0 is vector that represents the origin point. According to this formula, the measure takes 0 as one and only reference point. The similarity between two documents d_i and d_j is determined with respective to the angle between the two points when looking from the origin. To construct a new concept of similarity, it is possible to use more than just one point of reference. We may have a more accurate assessment of how close or distant a pair of points is, if we look at them from many different viewpoints. From a third point $d_{h,}$ the directions and distances to d_i and d_j are indicated respectively by the difference vectors *(d_i − d_h)* and *(dj − d_h)*. By standing at various reference points d_h to view $d_{i,}$ d_j and working on their difference vectors, we define similarity between the two documents as:

$$\sin_{d_i, d_j \varepsilon S_r}\ (d_i, d_j) = \frac{1}{n - n_r} \sum_{d_h \varepsilon S \setminus S_r} \sin\ (d_i - d_h, d_j - d_h) \qquad (10)$$

As described by the above equation, similarity of two documents d_i and d_j - 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. A presumption of cluster memberships has been made prior to the measure. 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 Multi-Viewpoint based Similarity, or MVS. Existing systems greedily picks the next frequent item set which represent the next cluster to minimize the overlapping between the documents that contain both the item set and some remaining item sets. In other words, the clustering result depends on the order of picking up the item sets, which in turns depends on the greedy heuristic. This method does not follow a sequential order of selecting clusters. Instead, we assign documents to the best cluster.

V. PROPOSED SYSTEM

The main work is to develop a novel hierarchal algorithm for document clustering which provides maximum efficiency and performance. A hierarchical algorithm clustering algorithm is based on the union between the two nearest clusters. The beginning condition is realized by setting every datum as a cluster. After a few iterations, it reaches the final clusters wanted. The final category of probabilistic algorithms is focused around model matching using probabilities as opposed to distances to decide clusters. It is particularly focused in studying and making use of cluster overlapping phenomenon to design cluster merging criteria. Proposing a new way to compute the overlap rate in order to improve time efficiency and "the veracity" is mainly concentrated. Based on the Hierarchical Clustering Method, the usage of Expectation-Maximization (EM) algorithm in the Gaussian Mixture Model to count the parameters and make the two sub-clusters combined when their overlap is the largest is narrated. Here, the data set is usually modeled with a fixed (to avoid overfitting) number of Gaussian distributions that are initialized randomly and whose parameters are iteratively optimized to fit better to the data set. This will converge to a local optimum, so multiple runs may produce different results. In order to obtain a hard clustering, objects are often then assigned to the Gaussian distribution they most likely belong to, for soft clustering this is not necessary.

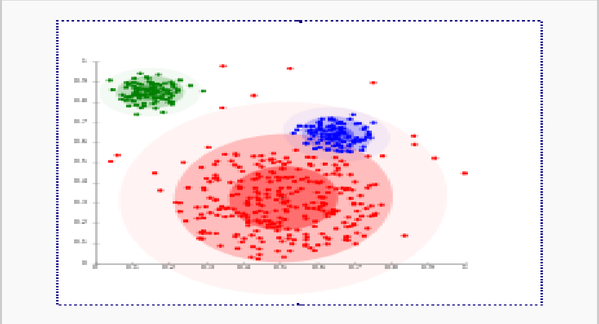

Fig.1 On Gaussian-distributed data, EM works well, since it uses Gaussians for modeling clusters

Experiments in document clustering data show that this approach can improve the efficiency of clustering and save computing time.

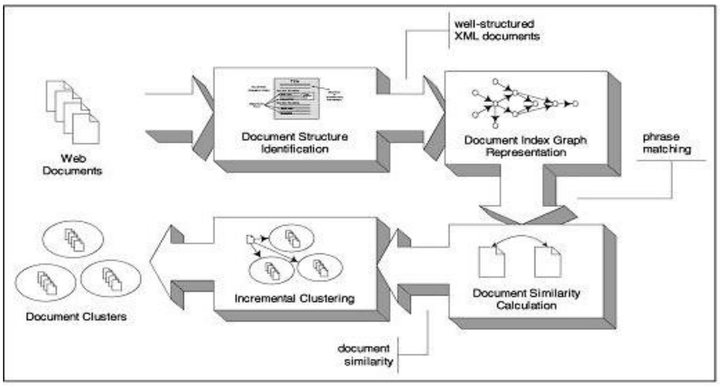

Fig. 2 System Architecture

Fig. 2 shows the basic system architecture. Given a data set satisfying the distribution of a mixture of Gaussians, the degree of overlap between components affects the number of clusters "perceived" by a human operator or detected by a clustering algorithm. In other words, there may be a significant difference between intuitively defined clusters and the true clusters mixture.

VI. CONCLUSION

The key contribution of this paper is the fundamental concept of similarity measure from multiple viewpoints. Theoretical analysis show that Multi-viewpoint based similarity measure (MVS) is potentially more suitable for text documents than the popular cosine similarity measure. The future methods could make use of the same principle, but define alternative forms for the relative similarity in or do not use average but have other methods to combine the relative similarities according to the different viewpoints. In future, it would also be possible to apply the proposed criterion functions for hierarchical clustering algorithms. It would be interesting to explore how they work types of sparse and high-dimensional data.

REFERENCES

1. A.K.Jain, M.N.Murty, P.J.Flynn," Data Clustering: A Review ", ACM Computing Surveys, Vol. 31, No. 3, pp. 265-321, Sept. 1999.

2. X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z., H. Zhou, M.Steinbach, D. JHand and D. Steinberg, "Top 10 algorithms in data mining," *Knowledge Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2007.

3. I. Guyon, U. von Luxburg, and R. C. Williamson, "Clustering: Science or Art?" *NIPS'09 Workshop on Clustering Theory*, 2009.

4. I. Dhillon and D. Modha, "Concept decompositions for large sparse text data using clustering," *Mach. Learn.*, vol. 42, no. 1-2, pp. 143–175, Jan 2001.

5.] S. Zhong, "Efficient online spherical K- means clustering," in *IEEE IJCNN*, 2005, pp. 3180–3185.

6. G. Salton. Automatic Text Processing. Addison-Wesley, New York, 1989.

7. M.J.McGill, "Introduction to Modern Information Retrieval", NY-1983.

8. Gerard M. Salton and Christopher Buckley, "Term Weighting Approaches in Automatic Text Retrieval", 24(5), pp. 513-523, 1988.

9. Paul Resnick, Mitesh Suchak, John Riedl, "Grouplens: an open architecture for collaborative filtering of netnews", CSCW- ACM conference on Computer supported cooperative work, pp. 175-186, 1994.

10. John Breese,David Heckerman,Carl kadie, " Empirical analysis of predictive algorithms for collaborative filtering", the 14th conference on uncertainty in Artificial Intelligence, pp. 43-52, 1998.

11.] Small, H., & Sweeney, E.," Clustering the Science Citation Index Using Co- Citations I", A Comparison of Methods. Scientometrics, 7, pp. 391- 409, 1985.

12. Jones, W. P., & Furnas, G. W., "Pictures of Relevance: A Geometric Analysis of Similarity Measures", Journal of the American Society for Information Science, 36 (6), pp. 420- 442, 1987.

13. Oberski, J., "Some Statistical Aspects of Co-Citation Cluster Analysis and a Judgment by Physicist", In A. F. J. van Raan (Ed.), Handbook of Quantitative Studies of Science & Technology, pp. 431-462, 1988.

14. Y. Zhao and G. Karypis, "Empirical and theoretical comparisons of selected criterion functions for document clustering," *Mach. Learn.*, vol. 55, no. 3, pp. 311–331, Jun 2004.

15.] R. O. Duda, P. E. Hart, and D. G. Stork, Pattern *Classification*, 2nd Ed. New York: John Wiley & Sons, 2001.

16. A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra, "Clustering on the unit hypersphere using von Mises-Fisher distributions," *J. Mach. Learn. Res.*,vol. 6, pp. 1345–1382, Sep 2005.

17. W. Xu, X. Liu, and Y. Gong, "Document clustering based on Nonnegative matrix factorization," in *SIGIR*, 2003, pp. 267–273.

18. C. D. Manning, P. Raghavan, and H. Sch ¨ utze, *An Introduction to Information Retrieval*. Press, Cambridge U., 2009.