Volume 6, Issue 2, February 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# Clustering With Multi View Point Based Similarity Measure

**Vasudha Rani Vaddadi***
IT Department, GMRIT, Rajam, Andhra Pradesh,
India

***Abstract— This 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. In this paper, we introduce a novel multi-viewpoint based similarity measure and two related clustering methods. The major difference between a traditional dissimilarity/similarity measure and ours is that the former uses 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, text mining, similarity measure.***

## I. INTRODUCTION

Clustering is the classification of objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often proximity according to some defined distance measure. Data clustering is a common technique for statistical data analysis, which is used in many fields, including machine learning, data mining, pattern recognition, image analysis and bioinformatics. The computational task of classifying the data set into k clusters is often referred to as k-clustering. Besides the term data clustering (or just clustering), there are a number of terms with similar meanings, including cluster analysis, automatic classification, numerical taxonomy, and typological analysis.

Document clustering aims to group, in an unsupervised way, a given document set into clusters such that documents within each cluster are more similar between each other than those in different clusters. It is an enabling technique for a wide range of information retrieval tasks such as efficient organization, browsing and summarization of large volumes of text documents. Cluster analysis aims to organize a collection of patterns into clusters based on similarity. Clustering has its root in many fields, such as mathematics, computer science, statistics, biology, and economics. In different application domains, a variety of clustering techniques have been developed, depending on the methods used to represent data, the measures of similarity between data objects, and the techniques for grouping data objects into clusters.

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. For this Hierarchical Clustering method provides a better improvement in achieving the result. Our project presents two key parts of successful Hierarchical document clustering. The first part is a document index model, the Document Index Graph, which allows for incremental construction of the index of the document set with an emphasis on efficiency, rather than relying on single-term indexes only. It provides efficient phrase matching that is used to judge the similarity between documents. This model is flexible in that it could revert to a compact representation of the vector space model if we choose not to index phrases. The second part is an incremental document clustering algorithm based on maximizing the tightness of clusters by carefully watching the pair-wise document similarity distribution inside clusters. Both the phases are based upon two algorithmic models called Gaussian Mixture Model and Expectation Maximization. The combination of these two components creates an underlying model for robust and accurate document similarity calculation that leads to much improved results in Web document clustering over traditional methods.

## II. CLUSTERING

Data clustering algorithms can be hierarchical. Hierarchical algorithms find successive clusters using previously established clusters. Hierarchical algorithms can be agglomerative ("bottom-up") or divisive ("top-down"). Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters. Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters. Partitional algorithms typically determine all clusters at once, but can also be used as divisive algorithms in the hierarchical clustering.

## HIERARCHICAL CLUSTERING

### Creating clusters

Hierarchical clustering builds or breaks up a hierarchy of clusters. The traditional representation of this hierarchy is a tree called as dendrogram, with individual elements at one end and a single cluster containing every element at the other. Agglomerative algorithms begin at the leaves of the tree, whereas divisive algorithms begin at the root. Cutting the tree at a given height will give a clustering at a selected precision. In the following example, cutting after the second row will yield clusters {a} {b c} {d e} {f}. Cutting after the third row will yield clusters {a} {b c} {d e f}, which is a coarser clustering, with a smaller number of larger clusters.

### Algorithmic steps for Agglomerative Hierarchical clustering

Let $X = \{x_1, x_2, x_3, ..., x_n\}$ be the set of data points.

1) Begin with the disjoint clustering having level L(0) = 0 and sequence number m = 0.

2) Find the least distance pair of clusters in the current clustering, say pair (r), (s), according to d[(r),(s)] = min d[(i),(j)] where the minimum is over all pairs of clusters in the current clustering.

3) Increment the sequence number: m = m +1.Merge clusters (r) and (s) into a single cluster to form the next clustering m. Set the level of this clustering to L(m) = d[(r),(s)].

4) Update the distance matrix, D, by deleting the rows and columns corresponding to clusters (r) and (s) and adding a row and column corresponding to the newly formed cluster. The distance between the new cluster, denoted (r,s) and old cluster(k) is defined in this way: d[(k), (r,s)] = min (d[(k),(r)], d[(k),(s)]).

5) If all the data points are in one cluster then stop, else repeat from step

Divisive Hierarchical clustering - It is just the reverse of Agglomerative Hierarchical approach.

For example, suppose this data is to be clustered, and the euclidean distance is the distance metric.

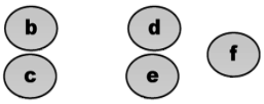




Fig 2.1 Raw data

The hierarchical clustering Dendogram would be as such:

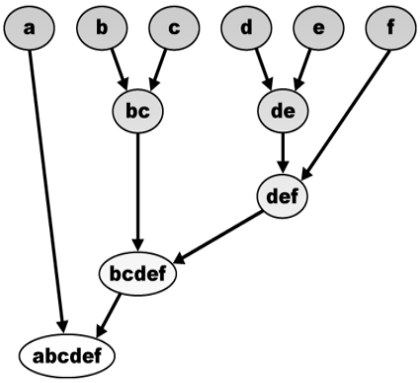



Fig 2.2 Traditional representation

This method builds the hierarchy from the individual elements by progressively merging clusters. In the example, six elements {a} {b} {c} {d} {e} and {f} are represented. The first step is to determine which elements to merge in a cluster. Usually, the major focus is to take the two closest elements, according to the chosen distance.

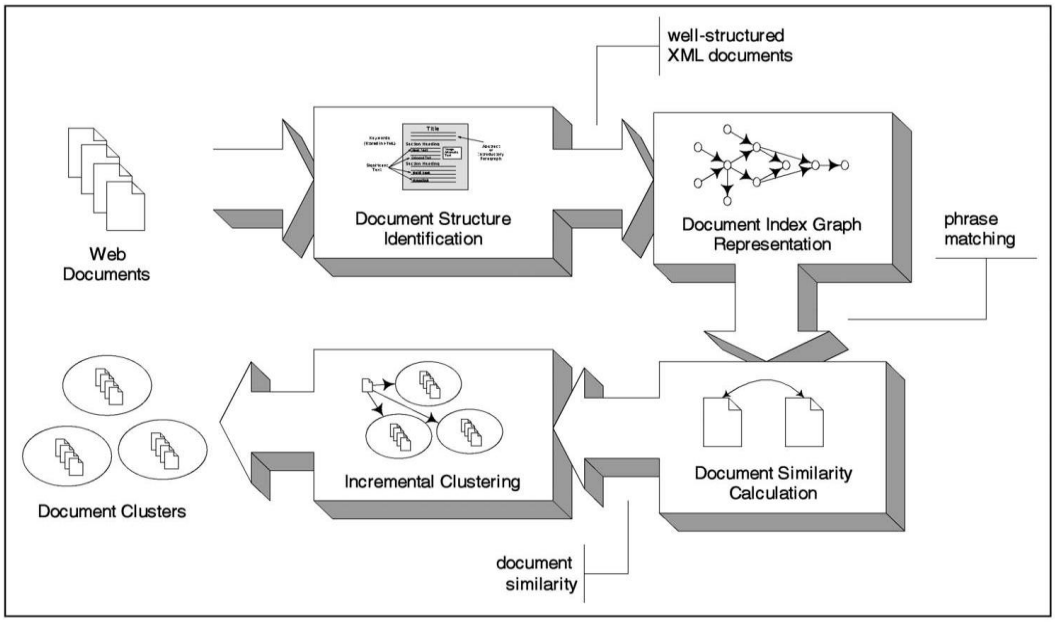
## III. SYSTEM ARCHITECTURE

The main work is to develop a novel hierarchal algorithm for document clustering which provides maximum efficiency and performance.

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.

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




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 corresponding to the components in the mixture.

## IV. MODULES

- HTML PARSER
- CUMMULATIVE DOCUMENT
- DOCUMENT SIMILARITY
- CLUSTERING

**HTML Parser**

- Parsing is the first step done when the document enters the process state.
- Parsing is defined as the separation or identification of meta-tags in a HTML document.
- Here, the raw HTML file is read and it is parsed through all the nodes in the tree structure.

**Cumulative Document**

- The cumulative document is the sum of all the documents, containing meta-tags from all the documents.
- We find the references (to other pages) in the input base document and read other documents and then find references in them and so on.
- Thus in all the documents their meta-tags are identified, starting from the base document.

**Document Similarity**

- The similarity between two documents is found by the cosine-similarity measure technique.
- The weights in the cosine-similarity are found from the TF-IDF measure between the phrases (meta-tags) of the two documents.
- This is done by computing the term weights involved.
- TF = C / T
- IDF = D / DF.

  D → quotient of the total number of documents
  DF → number of times each word is found in the entire corpus
  C → quotient of no of times a word appears in each document
  T → total number of words in the document

  **TFIDF = TF * IDF**

**Clustering**

- Clustering is a division of data into groups of similar objects.
- Representing the data by fewer clusters necessarily loses certain fine details, but achieves simplification.

The similar documents are grouped together in a cluster, if their cosine similarity measure is less than a specified threshold.

**Similarity Measures**

The concept of similarity is fundamentally important in almost every scientific field. For example, in mathematics, geometric methods for assessing similarity are used in studies of congruence and homothety as well as in allied fields such as trigonometry. Topological methods are applied in fields such as semantics. Graph theory is widely used for assessing cladistic similarities in taxonomy. Fuzzy set theory has also developed its own measures of similarity, which find application in areas such as management, medicine and meteorology. An important problem in molecular biology is to measure the sequence similarity of pairs of proteins.

A review or even a listing of all the uses of similarity is impossible. Instead, perceived similarity is focused on. The degree to which people perceive two things as similar fundamentally affects their rational thought and behavior. Negotiations between politicians or corporate executives may be viewed as a process of data collection and assessment of the similarity of hypothesized and real motivators. The appreciation of a fine fragrance can be understood in the same way. Similarity is a core element in achieving an understanding of variables that motivate behavior and mediate affect.
Not surprisingly, similarity has also played a fundamentally important role in psychological experiments and theories. For example, in many experiments people are asked to make direct or indirect judgments about the similarity of pairs of objects. A variety of experimental techniques are used in these studies, but the most common are to ask subjects whether the objects are the same or different, or to ask them to produce a number, between say 1 and 7, that matches their feelings about how similar the objects appear (e.g., with 1 meaning very dissimilar and 7 meaning very similar). The concept of similarity also plays a crucial but less direct role in the modeling of many other psychological tasks. This is especially true in theories of the recognition, identification, and categorization of objects, where a common assumption is that the greater the similarity between a pair of objects, the more likely one will be confused with the other. Similarity also plays a key role in the modeling of preference and liking for products or brands, as well as motivations for product consumption

## V. EXPERIMENTAL RESULTS

**Simple counting**

This can begin by counting the number of times each of the words appear in each of the documents,

| **Document 1** | | **Document 2** | | **Document 3** | |
|---|---|---|---|---|---|
| **Word** | **C** | **Word** | **C** | **Word** | **C** |
| airplane | 5 | Book | 3 | building | 6 |
| Blue | 1 | Car | 7 | Car | 1 |
| Chair | 7 | Chair | 4 | carpet | 3 |
| computer | 3 | Justice | 2 | ceiling | 4 |
| Forest | 2 | Milton | 6 | Chair | 6 |
| Justice | 7 | Newton | 3 | cleaning | 4 |
| Love | 2 | Pond | 2 | justice | 8 |
| Might | 2 | Rose | 5 | libraries | 2 |
| Perl | 5 | Shakespeare | 4 | newton | 2 |
| Rose | 6 | Slavery | 2 | Perl | 5 |
| Shoe | 4 | Thesis | 2 | Rose | 7 |
| Thesis | 2 | Truck | 1 | science | 1 |
| **Totals (T)** | 46 | **Totals (T)** | 41 | **Totals (T)** | 49 |

**Word Counting**

Given this simple counting method, searches for “rose” can be sorted by its “term frequency” (TF) — the quotient of the number of times a word appears in each document (C), and the total number of words in the document (T) — TF = C / T. In the first case, rose has a TF value of 0.13. In the second case TF is 0.12, and in the third case it is 0.14. Thus, by this rudimentary analysis, Document 3 is most significant in terms of the word “rose”, and Document 2 is the least. Document 3 has the highest percentage of content containing the word “rose”.

*Accounting for common words*

Unfortunately, this simple analysis needs to be offset considering frequently occurring terms across the entire corpus. Good examples are stop words or the word “human” in MEDLINE. Such words are nearly meaningless because they apear so often. Consider the table which includes the number of times each word is found in the entire corpus (DF), and the quotient of the total number of documents (D or in this case, 3) and DF — IDF = D / DF. Words with higher scores are more significant across the entire corpus. Search terms whose IDF (”inverse document frequency”) score approach 1 are close to useless because they exist in just about every document:

| **Document 1** | | | **Document 2** | | | **Document 3** | | |
|---|---|---|---|---|---|---|---|---|
| **Word** | **DF** | **IDF** | **Word** | **DF** | **IDF** | **Word** | **DF** | **IDF** |
| airplane | 1 | 3.0 | Book | 1 | 3.0 | building | 1 | 3.0 |
| blue | 1 | 3.0 | Car | 2 | 1.5 | car | 2 | 1.5 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| chair | 3 | 1.0 | Chair | 3 | 1.0 | carpet | 1 | 3.0 |
| computer | 1 | 3.0 | Justice | 3 | 1.0 | ceiling | 1 | 3.0 |
| forest | 1 | 3.0 | Milton | 1 | 3.0 | chair | 3 | 1.0 |
| justice | 3 | 1.0 | Newton | 2 | 1.5 | cleaning | 1 | 3.0 |
| love | 1 | 3.0 | Pond | 1 | 3.0 | justice | 3 | 1.0 |
| might | 1 | 3.0 | Rose | 3 | 1.0 | libraries | 1 | 3.0 |
| perl | 2 | 1.5 | Shakespeare | 1 | 3.0 | newton | 2 | 1.5 |
| rose | 3 | 1.0 | Slavery | 1 | 3.0 | Perl | 2 | 1.5 |
| shoe | 1 | 3.0 | Thesis | 2 | 1.5 | rose | 3 | 1.0 |
| thesis | 2 | 1.5 | Truck | 1 | 3.0 | science | 1 | 3.0 |

**DF AND IDF**

*TFIDF ANALYSIS*

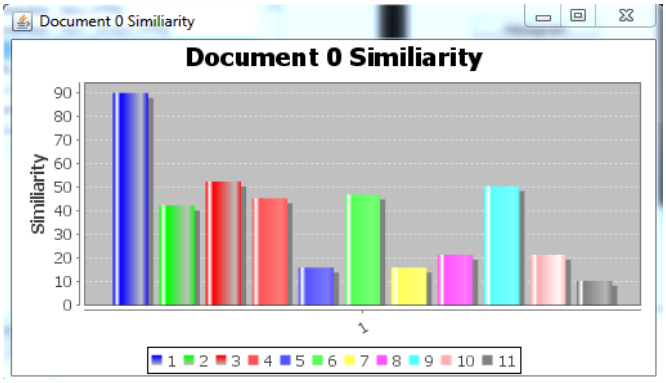
By taking into account these two factors term frequency (TF) and inverse document frequency (IDF) — it is possible to assign "weights" to search results and therefore ordering them statistically. Put another way, a search result's score ("ranking") is the product of TF and IDF:

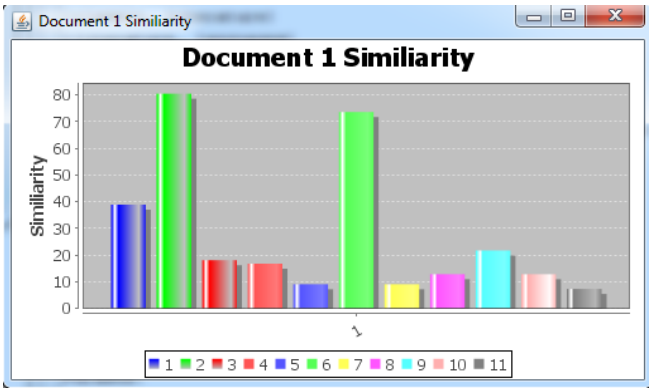
| Document 1 | | Document 2 | | Document 3 | |
|---|---|---|---|---|---|
| **Word** | **TFIDF** | **Word** | **TFIDF** | **Word** | **TFIDF** |
| airplane | 0.326 | Milton | 0.439 | building | 0.367 |
| shoe | 0.261 | shakespeare | 0.293 | ceiling | 0.245 |
| computer | 0.196 | Car | 0.256 | cleaning | 0.245 |
| perl | 0.163 | Book | 0.220 | carpet | 0.184 |
| chair | 0.152 | Pond | 0.146 | justice | 0.163 |
| justice | 0.152 | Slavery | 0.146 | perl | 0.153 |
| forest | 0.130 | Rose | 0.122 | rose | 0.143 |
| love | 0.130 | Newton | 0.110 | chair | 0.122 |
| might | 0.130 | Chair | 0.098 | libraries | 0.122 |
| rose | 0.130 | Thesis | 0.073 | newton | 0.061 |
| blue | 0.065 | Truck | 0.073 | science | 0.061 |
| thesis | 0.065 | Justice | 0.049 | car | 0.031 |

**Total calculation**

## HISTOGRAM FORMATION

After finding the similarity and the OLP Rate, Histogram is formed as follows



## VI. CONCLUSION & FUTURE WORK

Given a data set, the ideal scenario would be to have a given set of criteria to choose a proper clustering algorithm to apply. Choosing a clustering algorithm, however, can be a difficult task. Even ending just the most relevant approaches for a given data set is hard. Most of the algorithms generally assume some implicit structure in the data set. One of the most important elements is the nature of the data and the nature of the desired cluster. Another issue to keep in mind is the kind of input and tools that the algorithm requires. This report has a proposal of a new hierarchical clustering algorithm based on the overlap rate for cluster merging. The experience in general data sets and a document set indicates that the new method can decrease the time cost, reduce the space complexity and improve the accuracy of clustering. Specially, in the document clustering, the newly proposed algorithm measuring result show great advantages. The hierarchical document clustering algorithm provides a natural way of distinguishing clusters and implementing the basic requirement of clustering as high within-cluster similarity and between-cluster dissimilarity.

In the proposed model, selecting different dimensional space and frequency levels leads to different accuracy rate in the clustering results. How to extract the features reasonably will be investigated in the future work.

There are a number of future research directions to extend and improve this work. One direction that this work might continue on is to improve on the accuracy of similarity calculation between documents by employing different similarity calculation strategies. Although the current scheme proved more accurate than traditional methods, there are still rooms for improvement

## REFERENCES

[1] Sun Da-fei,Chen Guo-li,Liu Wen-ju. The discussion of maximum likehood parameter estimation based on EM algorithm. Journal of HeNan University. 2002,32(4):35~41

[2] Khaled M. Hammouda, Mohamed S. Kamel , efficient phrase-based document indexing for web document clustering , IEEE transactions on knowledge and data engineering, October 2004

[3] Jeff A. Bilmes. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. ICSI TR-97-021, U.C. Berkeley, 1998.

[4] Shi zhong, joydeep ghosh, Generative Model-Based Document Clustering: A Comparative Study, The University of Texas.