Volume 3, No. 11, November 2012
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

PERFORMANCE EVALUATION OF MULTIVIEWPOINT-BASED SIMILARITY MEASURE FOR DATA CLUSTERING

K.A.V.L.Prasanna[*1], Mr. Vasantha Kumar[*2]
[*1]Post Graduate Student, Avanthi Institute of Engineering and Technology, Visakhapatnam
chpraasanna48@yahoo.com
[*2]HOD, Dept of C.S.E, Avanthi Institute of Engineering and Technology, Visakhapatnam

Abstract: Some cluster relationship has to be considered for all clustering methods surrounded by the data objects which will be applied on. There may be a similarity between a pair of objects which can be defined as a choice of explicitly or implicitly. We in this paper introduce a novel multiviewpoint based similarity measure and two related clustering methods. The main distinctness of our concept with a traditional dissimilarity/similarity measure is that the aforementioned dissimilarity/similarity exercises only a single view point for which it is the base and where as the mentioned Clustering with Multiviewpoint-Based Similarity Measure uses many different viewpoints that are objects and are assumed to not be in the same cluster with two objects being measured. By utilizing multiple viewpoints, countless descriptive evaluation could be accomplished. In order to assist this declaration, the theoretical analysis and empirical study are carried. Depending on this new measure two criterion functions are proposed for document clustering. We examine them with certain distinguished clustering algorithms which use other preferred coincident measures on different group of documents in order to verify the improvement of our scheme.

Index terms: Data mining, text mining, similarity measure, multi-viewpoint similarity measure, clustering methods.

INTRODUCTION

Clustering is a process of grouping a set of physical or abstract objects into classes of *similar* objects and is a most interesting concept of data mining in which it is defined as a collection of data objects that are similar to one another. Purpose of Clustering is to catch fundamental structures in data and classify them into meaningful subgroup for additional analysis. Many of the clustering algorithms have been published every year and can be proposed for different research fields. They were developed by using various techniques and approaches. But according to the recent study k-means has been one of the top most data mining algorithms presently. For many of the practitioners k-means is the favorite algorithm in their related fields to use. Even though it is a top most algorithm, it has a few basic drawbacks when clusters are of differing sizes, densities and non-globular shape. Irrespective of the drawbacks is simplicity, understandability, and scalability is the main reasons that made the algorithm popular.

An algorithm with adequate performance and usability in most of application scenarios could be preferable to one with better performance in some cases but limited usage due to high complexity. 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 (CS) instead of Euclidean distance as the measure, is deemed to be more suitable .In, Banerjee et al. showed that Euclidean distance was indeed one particular form of a class of distance measures called Bregman divergences. They proposed Bregman hard clustering algorithm, in which any kind of the Bregman divergences could be applied. Kullback-Leibler divergence was a special case of Bregman divergences that was said to give good clustering results on document data sets. Kullback-Leibler divergence is a good example of nonsymmetrical measure. Also on the topic of capturing dissimilarity in data, Pakalska et al. found that the discriminative power of some distance measures could increase when their non-Euclidean and nonmetric attributes were increased. They concluded that non-Euclidean and nonmetric measures could be informative for statistical learning of data. In , Pelillo even argued that the symmetry and nonnegative assumption of similarity measures was actually a limitation of current state-of-the-art clustering approaches. Simultaneously, clustering

Still requires more robust dissimilarity or similarity measures; recent works such as [8] illustrate this need. The work in this paper is motivated by investigations from the above and similar research findings. It appears to us that the nature of similarity measure plays a very important role in the success or failure of a clustering method. Our first objective is to derive a novel method for measuring similarity between data objects in sparse and high-dimensional domain, particularly text documents. From the proposed similarity measure, we then formulate new clustering criterion functions and introduce their respective clustering algorithms, which are fast and scalable like k-means, but are also capable of providing high-quality and consistent performance.

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 (CS) instead of Euclidean distance as the measure, is deemed to be more suitable.

The work in this paper is motivated by investigations from the above and similar research findings. It appears to us that the nature of similarity measure plays a very important role in the success or failure of a clustering method. Our first objective is to derive a novel method for measuring similarity between data objects in sparse and high-dimensional domain, particularly text documents. From the proposed similarity measure, we then formulate new clustering criterion functions and introduce their respective clustering algorithms, which are fast and scalable like k-means, but are also capable of providing high-quality and consistent performance.

BACKGROUND WORK

Document clustering is one of the text mining techniques. It has been around since the inception of text mining domain. It is s process of grouping objects into some categories or groups in such a way that there is maximization of intra-cluster object similarity and inter-cluster dissimilarity. Here an object does mean a document and term refers to a word in the document. Each document considered for clustering is represented as an ***m*** – dimensional vector ***d***. The ***r***epresents the total number of terms present in the given document. Document vectors are the result of some sort of weighting schemes like TF-IDF (Term Frequency –Inverse Document Frequency). Many approaches came into existence for document clustering. They include information theoretic co-clustering, non – negative matrix factorization, and probabilistic model based method and so on. However, these approaches did not use specific measure in finding document similarity. In this paper we consider methods that specifically use certain measurement. From the literature it is found that one of the popular measures is Eucludian distance.

$$\text{Dist}(d_i, d_j) = \|d_i - d_j\|$$

K-means is one of the popular clustering algorithms in the world. It is in the list of top 10. Due to its simplicity and ease of use it is still being used in the mining domain. Euclidian distance measure is used in *k*means algorithm. The main purpose of the k-means algorithm is to minimize the distance, as per Euclidian measurement, between objects in clusters. The centroid of such clusters is represented as:

$$\text{Min } \Sigma k \Sigma \|di - Cr\|2 \qquad (2)$$

$$r=1\ di \in Sr$$

In text mining domain, cosine similarity measure is also widely used measurement for finding document similarity, especially for hi-dimensional and sparse document clustering. The cosine similarity measure is also used in one of the variants of ***k***-means known as spherical ***k***-means. It is mainly used to maximize the cosine similairity between cluster's centroid and the documents in the cluster. The difference between k-means that uses Euclidian distance and the k-means that make use of cosine similarity is that the former focuses on vector magnitudes while the latter focuses on vector directions. Another popular approach is known as graph partitioning approach. In this approach the document corpus is considered as a graph. Min – max cut algorithm is the one that makes use of this approach and it focuses on minimizing centroid function.
(3)

Other graph partitioning methods include Normalized Cut and Average Weight is used for document clustering purposes successfully. They used pair wise and cosine similarity score for document clustering. For document clustering analysis of criterion functions is made. CLUTO software package where another method of document clustering based on graph partitioning is implemented. It builds nearest neighbor graph first and then makes clusters. In this approach for given non-unit vectors of document the extend Jaccard coefficient is:

$$Sim_{ejacc}(u_i, u_j) = \frac{u_i u_j}{\|u_i\|2 + \|u_j\|2 - uti\,uj} \qquad (4)$$

$$Min \sum_{r=1}^{k} \frac{D_r^t D}{\|Dr\|^2}$$

Both direction and magnitude are considered in Jaccard coefficients when compared with cosine similarity and Euclidean distance. When the documents in clusters are represented as unit vectors, the approach is very much similar to cosine similarity. All measures such as cosine, Euclidean, Jaccard, and Pearson correlation are compared. The conclusion made here is that Ecludean and Jaccard are best for web document clustering. In [1] and research has been made on categorical data. They both selected related attributes for given subject and calculated distance between two values. Document similarities can also be found using approaches that are concept and phrase based. In [1] tree-milarity measure is used conceptually while proposed phrase-based approach. Both of them used an algorithm known as Hierarchical Agglomerative Clustering in order to perform clustering. Their computational complexity is very high that is the drawback of these approaches. For XML documents also measures are found to know structural similarity [5]. However, they are different from normal text document clustering.

MULTI-VIEWPOINT BASED SIMILARITY

Our main aim is to find similarity between documents or objects while performing clustering is multi-view based similarity. It makes use of more than one point of reference as opposed to existing algorithms used for clustering text

documents. As per our approach the similarity between two documents is calculated as:

$$Sim(di,dj) = 1/n\text{-}nr \Sigma Sim (di\text{-}dh, dj\text{-}dh) \quad (5)$$
$$dt,dj \in Srdh \in S\backslash Sr$$

Here the description of this approach can be given like this. Consider two point ***di*** and ***dj*** in cluster ***Sr***. The similarity between those two points is viewed from a point ***dh*** which is outside the cluster. Such similarity is equal to the product of cosine angle between those points with respect to Eucludean distance between the points. An assumption on which this definition is based on is "***dh*** is not the same cluster as ***di*** and ***dj***. When distances are smaller the chances are higher that the ***dh*** is in the same cluster. Though various viewpoints are useful in increasing the accuracy of similarity measure there is a possibility of having that give negative result. However the possibility of such drawback can be ignored provided plenty of documents to be clustered.

A series of algorithms are proposed to achieve MVS (Multi-View point Similarity). The following is a procedure for building similarity matrix of MVS.

a. procedure BUILDMVSMATRIX (A)
b. for r ← 1: c do
c. DSISrI←_di/∈ ΣSr di
d. nSISr ← |S ISr|
e. end for
f. for i ← 1 : n do
g. r ← class of di
h. for j ← 1 : n do
i. if dj ∈ Sr then
j. aij ← dti dj – dti DSISr nSISr – dt j DS\Sr nSISr + 1
k. else
l. aij←dti dj−dti DSISr –dj nSISr −1–dt j DSISr –dj nSISr −1
m. end if
n. end for
o. end for
p. return A = {aij}n×n
q. end procedure

Algorithm 3:Procedure for building MVS similarity matrix
From the consition it is understood that when di is considered closer to dl, the dl can still be considered being closer to di as per MVS. For validation purpose the following algorithm is used.

Require: 0 < percentage ≤ 1

a. procedure GETVALIDITY(validity,A, percentage)
b. for r ← 1 : c do
c. qr ← _percentage × nr
d. if qr = 0 then _ percentage too small
e. qr ← 1
f. end if
g. end for
h. for i ← 1 : n do
i. {aiv[1], . . . , aiv[n] } ←Sort {ai1, . . . , ain}
j. s.t. aiv[1] ≥ aiv[2] ≥ . . . ≥ aiv[n] {v[1], . . . , v[n]} ← permute {1, . . . , n}
k. r ← class of di
l. validity(di) ← |{dv[1], . . . , dv[qr] } ∩ Sr| qr
m. end for
n. validity ← _ni←1 validity(di)n
o. return validity
p. end procedure

Algorithm 4:Procedure for getting validity score
The final validity is calculated by averaging overall the rows of A as given in line 14. When the validity score is higher, the suitability is more for clustering.

SYSTEM DESIGN

The system design for finding similarity between documents or objects is as follows:

a. Initializing the weights parameters.
b. Using the EM algorithm to estimate their means and covariance.
c. Grouping the data to classes by the value of probability density to each class and calculating the weight of each class.
d. Repeat the first step until the cluster number reaches the desired number or the largest OLR is smaller than the predefined threshold value. Go to step 3 and output the result. A distinctive element in this algorithm is to use the overlap rate to measure similarity between clusters.

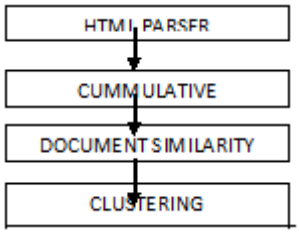


Figure 4.1 Design Layout

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

HTML Parser:

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

Cumulative Document:

a. The cumulative document is the sum of all the documents, containing meta-tags from all the documents.
b. 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.
c. 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.

a. The weights in the cosine-similarity are found from the TF-IDF measure between the phrases (meta-tags) of the two documents.
b. This is done by computing the term weights involved.
c. TF = C / T
d. IDF = D / DF.
e. D → quotient of the total number of documents
f. DF → number of times each word is found in the entire corpus
g. C → quotient of no of times a word appears in each document
h. T → total number of words in the document
i. **TFIDF = TF * IDF**

Clustering:

a. Clustering is a division of data into groups of similar objects.
b. Representing the data by fewer clusters necessarily loses certain fine details, but achieves simplification.
c. The similar documents are grouped together in a cluster, if their cosine similarity measure is less than a specified threshold [9].

Proposed architecture:

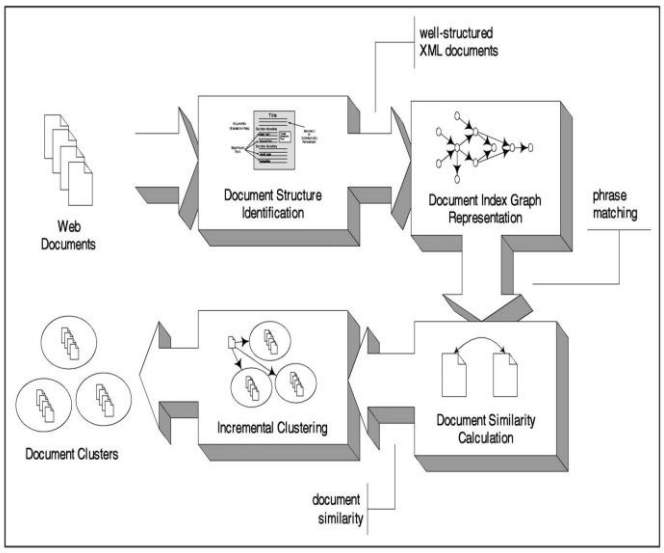


Figure 4.2 Architecture

Input design:

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

a. What data should be given as input?
b. How the data should be arranged or coded?
c. The dialog to guide the operating personnel in providing input.
d. Methods for preparing input validations and steps to follow when error occur.

Objectives:

a. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
b. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
c. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

Output design:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

a. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
b. Select methods for presenting information.
c. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

a. Convey information about past activities, current status or projections of the
b. Future.
c. Signal important events, opportunities, problems, or warnings.
d. Trigger an action.
e. Confirm an action.

IMPLEMENTATION

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays

the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors. An actor is represents a user or another system that will interact with the system you are modeling. A use case is an external view of the system that represents some action the user might perform in order to complete a task.

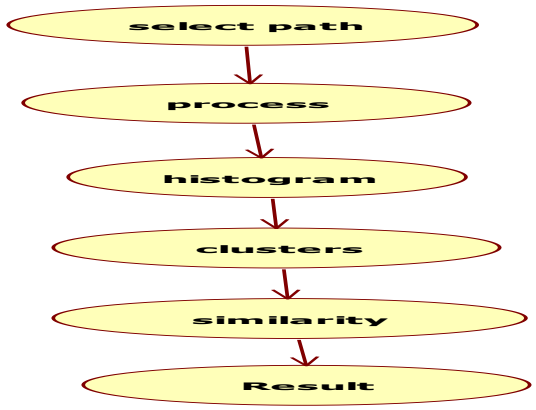


Figure 5.1 Use case diagram

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation. These perspectives become evident as the diagram is created and help solidify the design. Class diagrams are arguably the most used UML diagram type. It is the main building block of any object oriented solution.

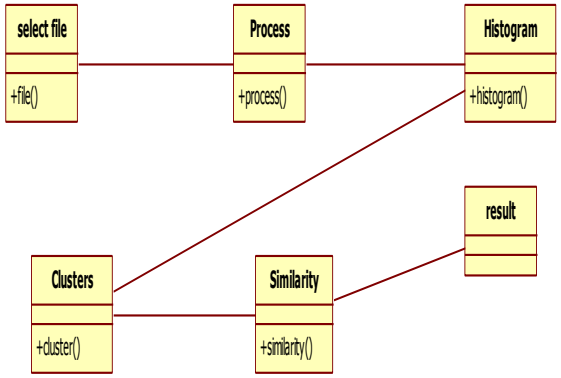


Figure 5.2: Class diagram

It shows the classes in a system, attributes and operations of each class and the relationship between each class. In most modeling tools a class has three parts, name at the top, attributes in the middle and operations or methods at the bottom. In large systems with many classes related classes are grouped together to to create class diagrams. Different relationships between diagrams are show by different types of Arrows. Below is a image of a class diagram. Follow the link for more class diagram examples.

Sequence diagrams in UML shows how object interact with each other and the order those interactions occur. It's important to note that they show the interactions for a particular scenario. The processes are represented vertically and interactions are show as arrows. This article explains thepurpose and the basics of Sequence diagrams.

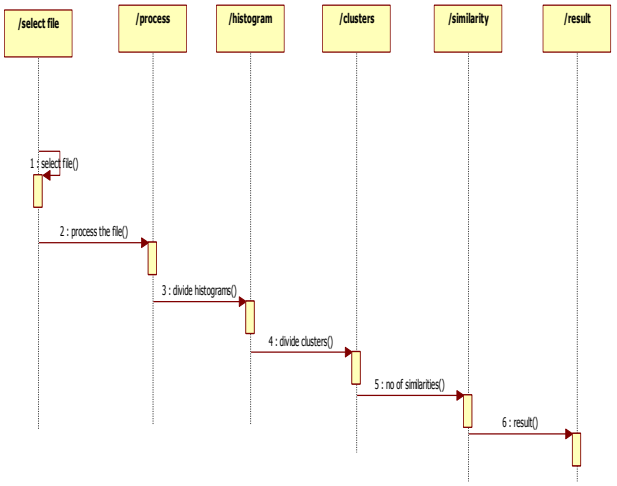
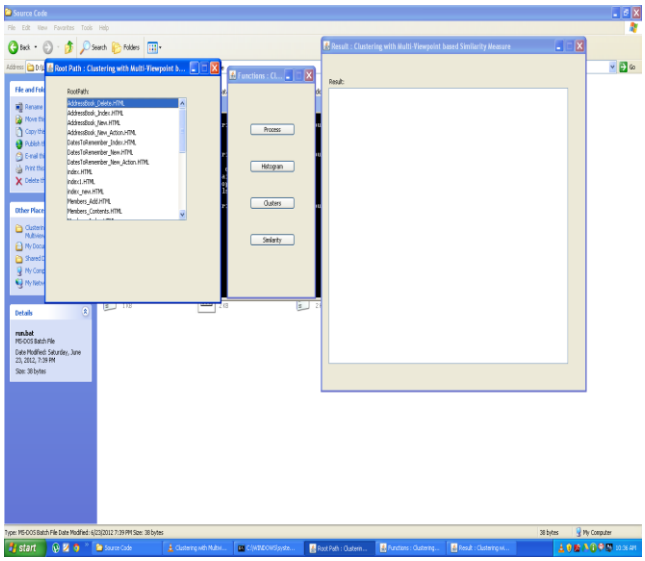
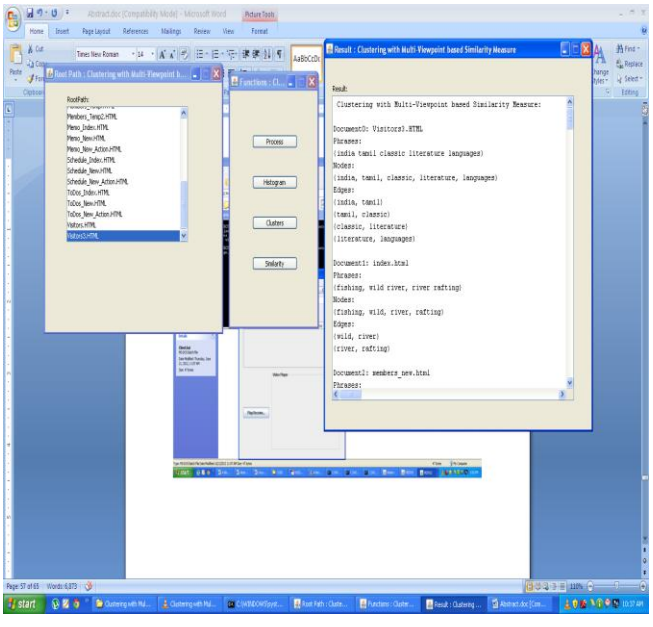
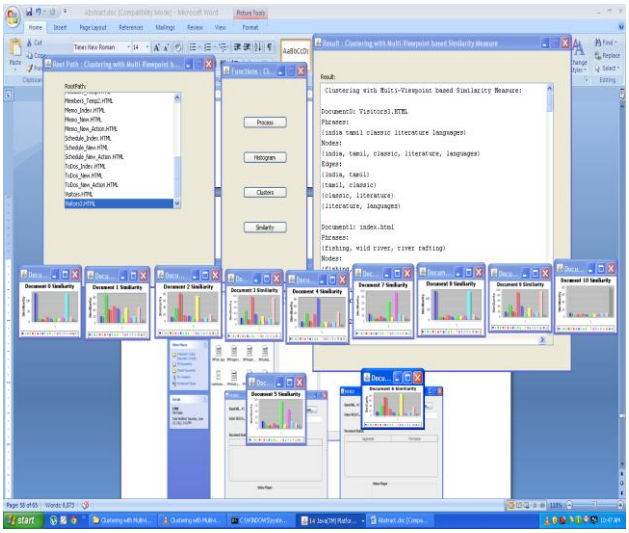


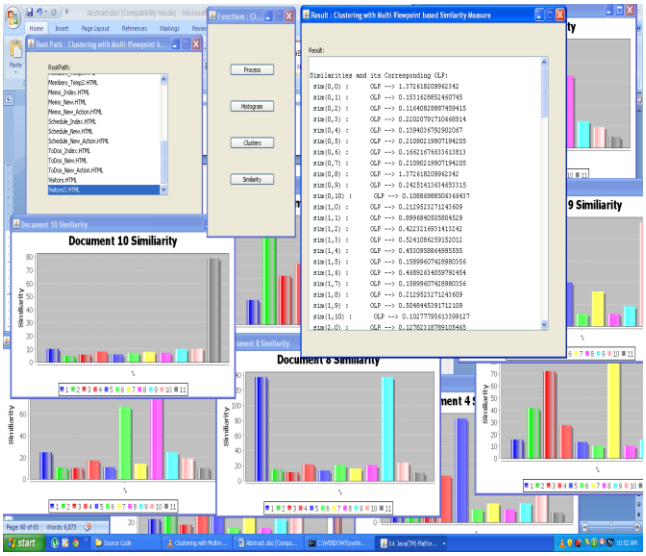
Figure 5.3: Sequence diagram

RESULTS









CONCLUSION

We in this paper propose a Multiviewpoint-based Similarity measuring method, named MVS.Both the Theoretical analysis and empirical examples represents that MVS is likely more supportive for text documents than the famous cosine similarity. Two criterion functions, IR and IV and the corresponding clustering algorithms MVSC-IR and MVSC-IV have been introduced in this paper. The proposed algorithms MVSC-IR and MVSC-IV shows that they could afford significantly advanced clustering execution ,when compared with other state-of-the-art clustering methods that use distinctive methods of similarity measure on a very large number of document data sets concealed by various assessment metrics. The main aspect of our paper is to introduce the basic concept of similarity measure from multiple viewpoints. This paper also concentrates on partitional clustering of documents. Further the proposed criterion functions for hierarchical clustering algorithms would also be achievable for applications .At last we have shown the application of MVS and its clustering algorithms for text data.

REFERENCES

[1]. Clustering with Multiviewpoint-Based Similarity Measure Duc Thang Nguyen, Lihui Chen, Senior Member, IEEE, and Chee Keong Chan, IEEE transactions on knowledge and data engineering, vol. 24, no. 6, june 2012

[2]. I. Guyon, U.V. Luxburg, and R.C. Williamson, "Clustering: Science or Art?," Proc. NIPS Workshop Clustering Theory, 2009.

[3]. I. Dhillon and D. Modha, "Concept Decompositions for Large Sparse Text Data Using Clustering," Machine Learning, vol. 42, nos. 1/2, pp. 143-175, Jan. 2001.

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

[5]. A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh, "Clustering with Bregman Divergences," J. Machine Learning Research, vol. 6, pp. 1705-1749, Oct. 2005.

[6]. E. Pekalska, A. Harol, R.P.W. Duin, B. Spillmann, and H. Bunke, "Non-Euclidean or Non-Metric Measures Can Be Informative," Structural, Syntactic, and Statistical Pattern Recognition, vol. 4109, pp. 871-880, 2006.

[7]. M. Pelillo, "What Is a Cluster? Perspectives from Game Theory," Proc. NIPS Workshop Clustering Theory, 2009.

[8]. D. Lee and J. Lee, "Dynamic Dissimilarity Measure for Support Based Clustering," IEEE Trans. Knowledge and Data Eng., vol. 22, no. 6, pp. 900-905, June 2010.

[9]. A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra, "Clustering on the Unit Hypersphere Using Von Mises-Fisher Distributions," J. Machine Learning Research, vol. 6, pp. 1345-1382, Sept. 2005.

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

[11]. I.S. Dhillon, S. Mallela, and D.S. Modha, "Information-Theoretic Co-Clustering," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 89-98, 2003.

[12]. http://iosrjournals.org/iosr-jce/papers/Vol4-issue6/F0463742.pdf

