ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

Efficient Clustering Algorithms in Text Mining

[1]Nataraj Gudapaty, [2]G Loshma, [3]Dr. Nagaratna P Hegde
[1,2,3]Dept. of CSE, Sri Vasavi Engineering College, Tadepalligudem, India
[3]Dept. of CSE, Vasavi College of Engineering Hyderabad, AP, India

Abstract
Text Mining is to process unstructured (textual) information, extract meaningful numeric indices from the text, and, thus, make the information contained in the text accessible to the various data mining (statistical and machine learning) algorithms. Information can be extracted to derive summaries for the words contained in the documents. Document clustering is a fundamental task of text mining, by which efficient organization, navigation, summarization, and retrieval of documents can be achieved. The clustering of documents presents difficult challenges due to the sparsity and the high dimensionality of text data, and to the complex semantics. K-means and PAM (partitioning around mediods) algorithms of text clustering and semantic-based vector space model, a semantic based PAM text clustering model is proposed to solve the problem on high-dimensional and sparse characteristics of text data set. The model reduces the semantic loss of the text data and improves the quality of text clustering. We propose a novel adaptive kernel K-means clustering algorithm and PAM (Partition Around Mediods) algorithm to combine textual content and citation information for clustering.
In this text mining process using semantics the comparison between K-Means and PAM is done. The time and space complexities of these two algorithms are compared and presented as bar charts and line charts using graphs.

Keywords
Document Clustering, Text Mining, Similarity Measure

I. Introduction

Text Mining is a flourishing new field that attempts to draw meaningful information from language text. In other words it is the process of analyzing text that is useful for particular purposes [1]. Clustering is a traditional data mining technique that attempts to organize un-clustered text documents into groups or clusters of text document in such a way that the clusters exhibit high intra cluster similarity and low inter cluster similarity. In general, text clustering methods try to separate the documents into groups in which each group represents a topic that is different from the other groups [12]. Decision trees [3], clustering based on data summarization [4], rule-based systems [5], statistical analysis [6], neural nets [7] are some of the methods that are used for text clustering. The most important aspect in text mining is that the output of the clustering algorithm depends on the features that have been selected [8]. Furthermore, the result of the clustering algorithm is based on the weights of the selected features. The Vector Space Model [9-10] is widely used document clustering method and represents data for text classification and clustering. The terms in the document is represented as a feature vector. The terms can be words or phrases. Each feature vector is assigned a term weight based on the term frequency of the terms in the documents. Similarity measures that rely on the feature vector is used to find the similarity between the documents (Cosine measures and the Jaccard measure). Generally in text mining techniques, we compute the term frequency of the terms in the document to find the importance of the term in the document. On the other hand, two terms can have the same term frequency in their documents, yet the meaning contributed by one term is more suitable to the meaning of the sentence than the meaning contributed by the other term. Hence, in the proposed model, the semantic structure of each term is captured rather than the frequency of the term within the document only. In this model, the concepts are analyzed on the sentence, document and corpus level. A concept-based similarity is used to determine the similarity among the documents and is based on the outcomes of the concept analysis on the sentence, document and corpus levels. Each sentence in a document is labeled by a semantic role labeler. This labeler determines the terms that contribute to the semantics of the sentence. Any term that has a semantic role in a sentence is called a concept. These concepts can be words or phrases and are dependent on the semantics of the sentence.

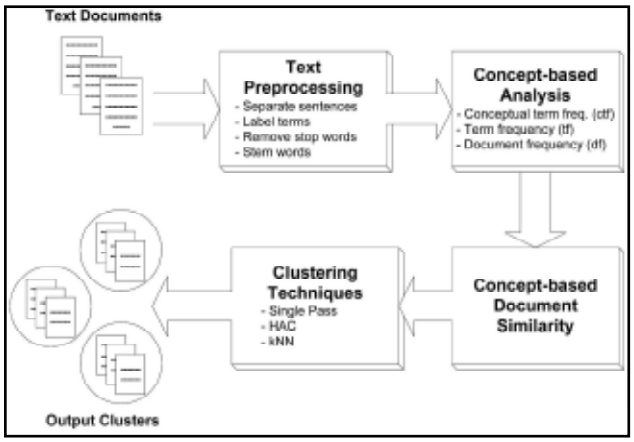

Fig. 1: The Concept Based Text Mining Model

With the exponential growth of online document content, data mining techniques for document clustering and classification have craved importance. Most traditional approaches performing document clustering do not consider the semantic relationship between the words. Thus if two documents talking about the same topic do that using different words (which may be synonyms), these algorithms cannot find the similarity between them and may cluster them into two different clusters. A simple solution to this problem is to use a knowledge base to enhance the document representation. Topic detection is a problem very closely related to that of document clustering. Intuitively, the goal here is to determine the set of all topics that are contained in a document. A document is a set of keywords. A topic is not necessarily the same as a keyword, but can be defined by a set of related keywords. The problem of topic detection therefore reduces to finding sets of related keywords in a document collection. Sets of such topics represent a cluster.
The rest of the paper is organized as follows: Section II, talks about the Existing System work. Section III, Proposed work of the text clustering using PAM algorithm, Section IV, describes Experimental results Finally, we conclude our paper and present the future work in Section V.

II. Existing System: The Implementation of K-Means Clustering Algorithm

In this section, we discuss about the Semantic based Text clustering and K-means algorithm.

A. Semantic Based Text Clustering

Conventional text representation models focus on whether a document contains specific keywords, or their appearance frequencies. For example, in the Vector Space Model (VSM) [9], documents are represented by vectors containing the frequency of all possible words (features) in a document set .Since many words rarely occur in a particular document, many of these features will have zero or low frequencies. Therefore, features are selected to represent documents according to their importance as dictated by criteria such as Document Frequency- Inverse Document Frequency, Information Gain, Mutual Information, a 2 c –test, and Term Strength [11]. Moreover, before applying feature selection, a common practice is to reprocess text by removing stop-words and applying word-stemming algorithms. Stop-words, such as the, and, and a, are believed to have no significance in capturing meaningful information. Word-stemming algorithms convert different word forms into a similar canonical form. Two popular stemming algorithms are used; the Porter stemmer [7], and using a lexicon dictionary lookup, such as Word Net [11].

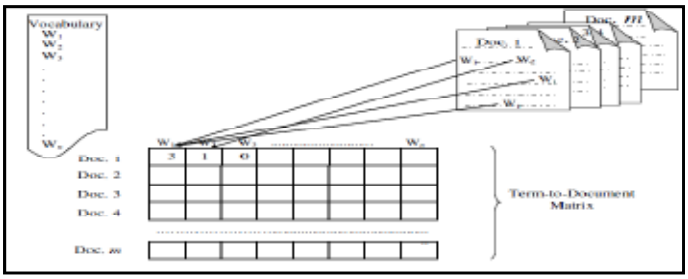

Fig. 2: The Vector Space Model

Document clustering aims to automatically divide documents into groups based on similarities of their contents. Each group (or cluster) consists of documents that are similar between themselves (have high intra-cluster similarity) and dissimilar to documents of other groups (have low inter-cluster similarity) .Clustering documents can be considered as an unsupervised task that attempts to classify documents by discovering underlying patterns, i.e., the learning process is unsupervised, which means that no need to define the correct output (i.e., the actual cluster into which the input should be mapped to) for an input. Document clustering is used to disambiguate results of information retrieval systems, by displaying them into specific topics. Aside from visualization of search results, it is used for taxonomy design and similarity search. Topic taxonomy (e.g., Yahoo!, and Open Directory dmoz. org) are constructed manually, but this process can be assisted by clustering a large samples of documents. Clustering can also help speed up similarity search, where close-by documents are to be retrieved additionally; in [6] it is argued that sentence-based text clustering can be a key factor for performance improvement of automatic speech recognition systems. There are many clustering techniques in the literature, each adopting a certain strategy for detecting the grouping in the data, such as K-means algorithm [9], Expectation Maximization [2] and hierarchical clustering [4], and many others surveyed in [7]. They can be divided into three main categories; partitioning, geometric, and probabilistic [1]. The following subsections report some algorithmic approaches under their perspective categories.

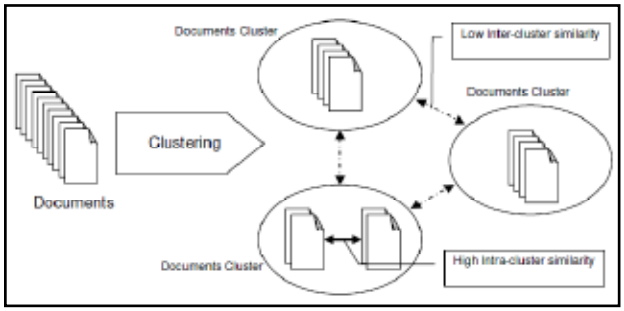

Fig. 3: K-Means Algorithm for Semantic Text Base Clustering

The basic idea of K-means is: each cluster is represented by the average value of all objects of the cluster, and then each sample is assigned to the least distance cluster by calculating the Euclidean distance to each of the Centers, and recalculates the Center's coordinate. Repeat the steps if the new Center set is different from the previous, then reassign, otherwise stop. K-means clustering algorithm is divided into three phases: In the first phase, text data is represented using semantic based vector space model, and the text similarity is calculated by the cosine similarity formula, that is,

$$sim(d_i, d_j) = \frac{\sum_{k=1}^{n} (w_{ik} * w_{jk})}{\sqrt{\sum_{k=1}^{n} w_{ik}^2 * \sum_{k=1}^{n} w_{jk}^2}}$$

For the text set $S_n = \{d_1, d_2, \cdots, d_n\}$, the similarity of each text object is calculated, as shown in equation and then the average of n text objects is calculated, as in the equation

$$s_i = \frac{1}{n} \sum_{j=1}^{n} sim(d_i, d_j)$$

$$s' = \frac{1}{n} \sum_{i=1}^{n} s_i$$

During the second phase, the number of clustering k is determined using the min-max principle. Assume that $i-1$ cluster centers have been selected, M is the center points set that have been selected, the selection methods of the next cluster center m_i is:

$$D_{\min} = \min\{\max\{d(x_i, q_j), q_j \in M\}, x_i \in S_n \setminus M\}$$

$|M| = i-1$, until the points' number is m. $D_{\min}$ of each of the m points selected is calculated, and then D_L is calculated using the equation , and the point of the largest D_L value is found out. Since the greater the depth D_L of each point, the larger fluctuations $D_{\min}$ in the point, the largest points of the D_L value shall be final number of clusters k, namely, the location i of this point shall be the desired k.

$$D_L(i) = D_{\min}(i) + |D_{\min}(i-1) - D_{\min}(i+1)|$$

In the third phase, the results in the second phase is used as the number of clusters for clustering algorithm at this stage, K-means clustering algorithm is used to cluster in order to obtain better clustering results.

III. Proposed Work Done

In this section, we discuss the Partition Around Medoid Clustering Algorithm. PAM is a clustering algorithm which groups a data set of n objects into k clusters by attempting to minimize the distance between the points in each cluster and the centre of that cluster, called medoid. A set of k medoids are randomly selected initially

and all other points are associated with their closest medoid. The most optimal clusters are identified through an iterative process where each medoid is compared to all non medoid points and swapped when the overall configuration is better. The PAM algorithm can be divided into three steps

Step 1: Build: choose optimal k medoids from T objects Step 2: Swap: The most intensive part of algorithm. Calculate cost: new_distance – old_distance for each swap of one medoid with another object.

Step 3: Evaluation: if the cost is negative, accept the swap with the best cost and go to step 2; otherwise record the medoids and terminate the program. An initial analysis of the original PAM implementation shows possible drawbacks of its legacy design. The R's environment open architecture has allowed the community to extend the language and add many new packages, making it de facto the standard tool among statisticians. However, R was not designed as a high performance language and although very popular it has slowly started to hit its limits. One of these is in-memory processing, where all the data to be processed in an R script must be loaded into memory. This is increasingly an issue since technology advances in measurement equipment and methodology, especially in the field of biological sciences has allowed scientists to gather data volumes that easily exceed the internal memory on modern workstations and PCs.

Algorithm PAMC_CWCD
input : training data set D
output : classification model
begin
//k ≥_number of classes in D
partition training data set D into k partitions (clusters) using PAM and find Med_i, i = 1..k
for each partition C_i, i = 1 …k
//total objects in each partition
Compute t_i
//compute class distribution
for each class c_j, j= 1…n
// total number of objects in each
//partition C_i and for each class label c_j
compute s_{ij};
// probability that an object in partition C_i
// belongs to class label c_j;
compute $p_{ij} = s_{ij} / t_i$;
end for;
end for;
set of Med_i and p_{ij} form classifier model;
end .

Fig. 2: PAMC_CWCD algorithm

Algorithm PAMC_classify
input : classification model and unknown O_u
output : predicted class of O_u

begin
for i = 1 …k
// distance of unknown object from each medoid
compute $d_i = d(O_u, Med_i)$
end for
for each c_j, j=1…n
for each partition C_i : i=1…k
compute $sum_i = \sum p_{ij} / d_i$
end for
end for
// track i for which sum_i is maximum
compute max (sum_i)
// return i^{th} class label for which sum_i is maximum
return c_i
end.

IV. Results

In order to implement these three algorithms on the basis on time complexity and space complexity.

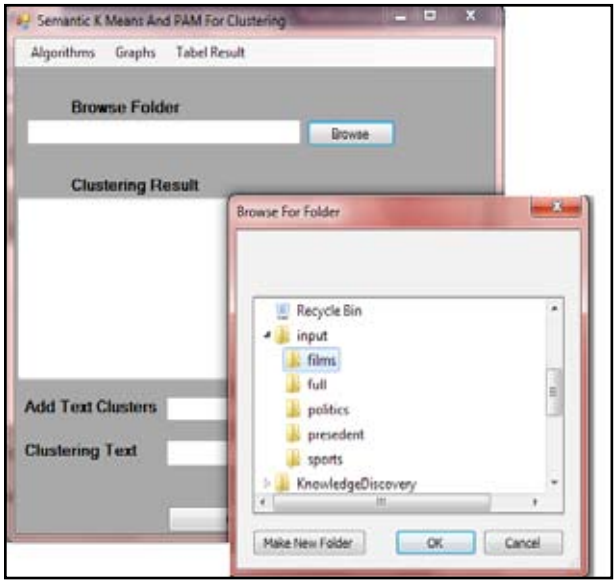

Fig. 4: Read Text Data Set from the File

Read the text data sets from the folders given to the input to the algorithms for K-means and PAM.

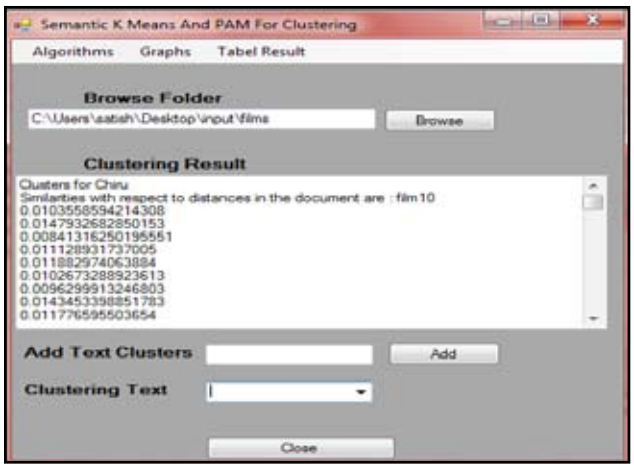

Fig. 5: Parse the Text Mining and Find out Cluster Item distances of the Cclustroids

Given to the data sets and cluster words to the algorithms and K-means and PAM algorithms calculate the distance of the clustroids

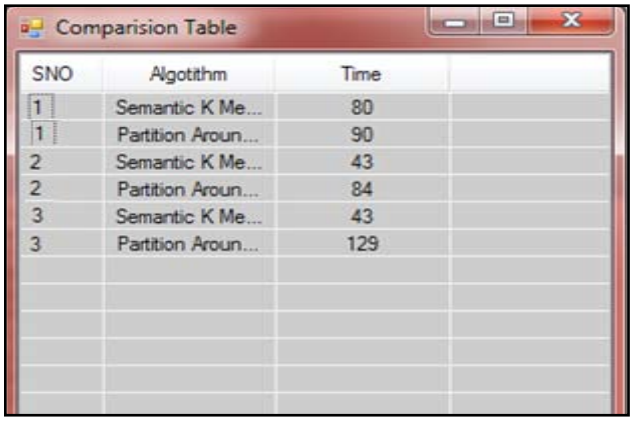

Fig. 5: Table for Display the Algorithms for Time Complexity

It display the how any times runs the algorithms with the given input and each time algorithms take the time complexity and display the time for each algorithm and display the time.

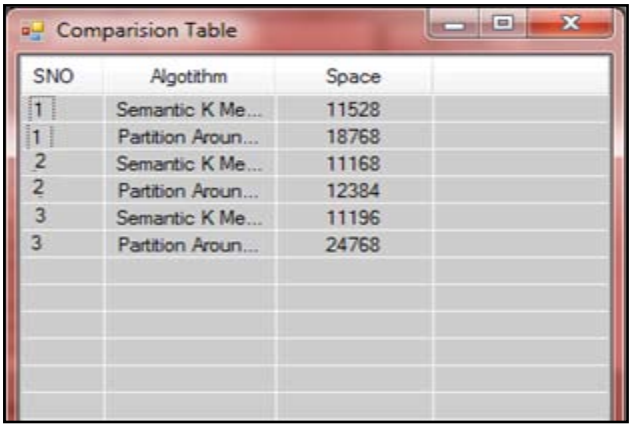

Fig. 6: Table for Display the Algorithms for Space Complexity

It display the how any times runs the algorithms with the given input and each time algorithms take the Space complexity and display the Space for each algorithm and display the Space.

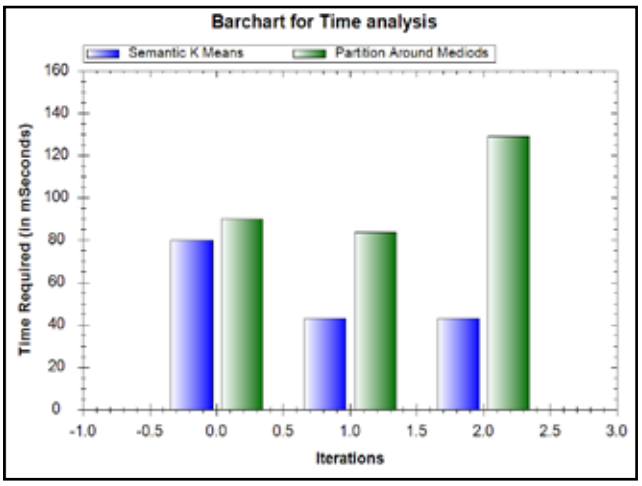

Fig. 7: Bar Chart on basis on Time Complexity

It display the bar chart representation for K-means and PAM Algorithm for Time complexity

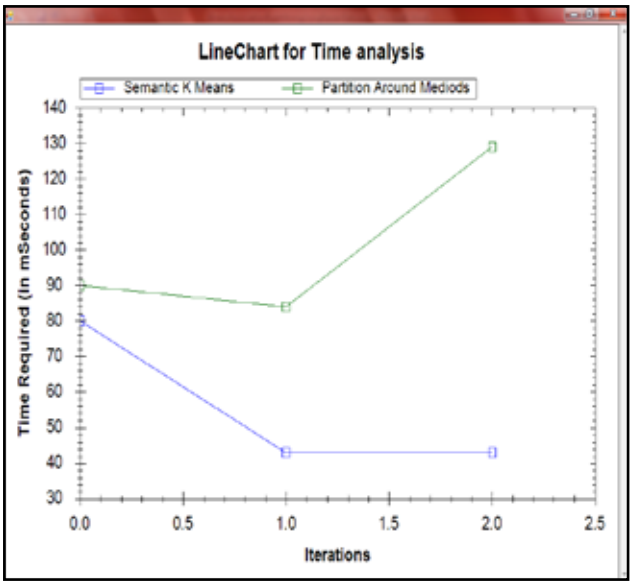

Fig. 8: Line Chart on basis on Time Complexity

It display the Line chart representation for K-means and PAM Algorithm for Time complexity

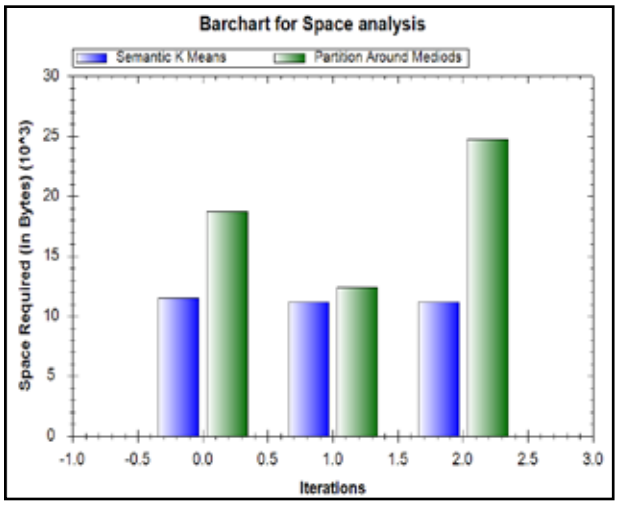

Fig. 9: Bar Chart on basis on Space Complexity

It display the bar chart representation for K-means and PAM Algorithm for Space complexity

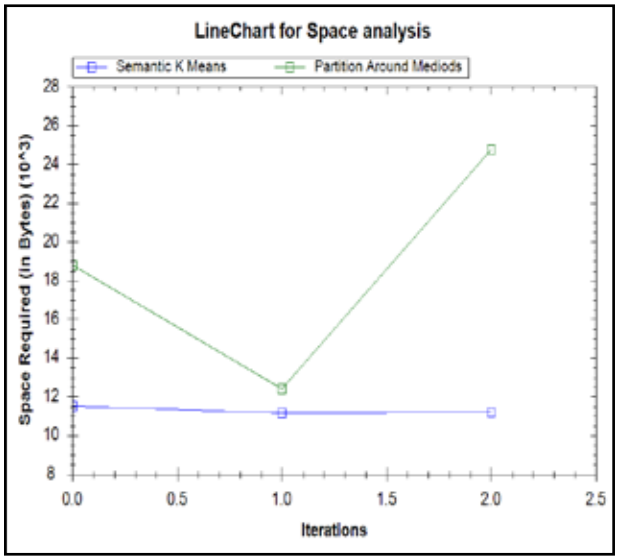

Fig. 10: Line Chart on basis on Space Complexity

It display the Line chart representation for K-means and PAM Algorithm for Space complexity

V. Conclusions

In this section we finally discuss about the semantic text clustering using the two algorithms Semantic K-Means and Portion around Algorithm (PAM). The time and space complexities of these two algorithms are compared and presented as bar charts and line charts using graphs. We have generated clusters and computed the results time complexity and space complexity in the presence of Text clustering for two algorithms. We have concluded K-Means algorithm takes less time and space compare to PAM.

References

[1] Berkhin, P.,"Survey of Clustering Data Mining Techniques", Technical Report, Accrue Software, 2002.

[2] Chakrabarti, S.,"Mining the Web: Discovering Knowledge from Hypertext Data", Morgan Kaufmann Publishers, 2003.

[3] Cios, K., Pedrycs, W., Swiniarski, R.,"Data Mining Methods for Knowledge Discovery", Kluwer Academic Publishers, 1998.
[4] Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.,"GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications", Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). Philadelphia, July 2002.
[5] Cutting, D.; Karger, D.; Pedersen, J.; Tukey, J.,"Scatter/gather: A cluster-based approach to browsing large document collections", In 16th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 126-135, 1993.
[6] Dasarathy, B.V.,"Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques", McGraw- Hill Computer Science Series. IEEE Computer Society Press, Las Alamitos, California (1991).
[7] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer T. K., Harshman, R.,"Indexing by Latent Semantic Analysis", Journal of the American Society for Information Science, 1990.
[8] Dempster,A.P., Laird N.M., land Rubin,"Maximum Likelihood from. Incomplete Data via the EM Algorithm", Jounal of the Royal Statistical Society, Series B (Methodological), 39(1), pp. 1-38, 1977.
[9] Eikvil, L.,"Information Extraction from World Wide Web – A Survey", Technical Report 945, Norwegian Computing Center, July, 1999.
[10] Hartigan, J. A., Wong, M. A.,"A K-means clustering algorithm". Applied Statistics, 28, pp. 100-108, 1979.
[11] Hasan, M., Matsumoto, Y.,"Document Clustering: Before and After the Singular Value Decomposition", Sapporo, Japan, Information Processing Society of Japan (IPSJ-TR: 99-NL-134.) pp. 47-55, 1999.
[12] 12. Hill, D.R.,"A vector clustering technique", In Samuelson, ed.: Mechanized Information Storage, Retrieval and Dissemination. North-Holland, Amsterdam (1968).