Hybrid Distance Based Document Clustering with Keyword and Phrase Indexing

K.Subhadra[1], Prof. M.Shashi[2]

**[1]Department of CSE,GIT,GITAM University,
Visakhapatnam, AP, India**

**[2]Department of CSSE,College Of Engineering
Andhra University, Visakhapatnam, AP, India**

Abstract

Document Clustering algorithms group a set of documents into subsets or *clusters*. Several applications of clustering exist in information retrieval. Our proposed method uses Scatter-Gather approach for clustering group of documents from an entire collection. The selected groupsare merged and the resulting set is again clustered. This process is repeateduntil a cluster of interest is found. This research presents a model for documentclustering that arranges unstructured documents into content-basedhomogeneous groups. The clustering approach uses the popular Cosine similarity measure combined with Euclidian distance measure. To the best of our knowledge, much work has been carried on keyword based clustering and Phrase index based clustering. Our method attempts to combine the two. The method has been applied to standard NewsGroup-20 dataset having documents distributed over 20 different topics. Results have been verified considering fixed number of clusters and different corpora and with variable number of clusters for fixed corpora. Both results indicate a steady increase in the overall purity of clustering compared to the keyword-based clustering method. With Keyword-based clustering, the purity was seen to increase for increasing number of clusters for a fixed corpora, but the purity was observed to decrease with fixed number of clusters and increase in number of corpora. In our method, the increase in purity was more pronounced with increase in number of clusters.

Keywords:*Document clustering, Phraseindex, Purity*

1. Introduction

Document clustering is a fundamental task of textmining by which efficient organization, navigation, summarization, and retrieval of documents can be achieved. Document clustering seeks to automatically partitionunlabeled documents into groups. Ideally, such groupscorrespond to genuine themes, topics, or categories ofthe corpus[1].The key input to any clustering algorithm is the distance measure. The distance measure is an importantmeans by which we can influence the outcome of clustering. Several applications of clustering exist in information retrieval. With regard to Newsgroups, two categories of clustering are in popular use. Scatter-Gather clusters the whole collection to getgroups of documents that the user can select or *gather*. The selected groupsare merged and the resulting set is again clustered. This process is repeateduntil a cluster of interest is found. As an alternative to the user-mediated iterative clustering in Scatter-Gather,we can also compute a static hierarchical clustering of a collection that isnot influenced by user interactions .Google News and its precursor, the Columbia News Blaster system, are examplesof this approach. In the case of news, we need to frequently recomputethe clustering to make sure that users can access the latest breakingstories. Clustering is well suited for access to a collection of news storiessince news reading is not really search, but rather a process of selecting asubset of stories about recent events. We can define the goal in hard flat clustering as follows. Given (i) a set of documents$D = \{d1, \ldots, dN\}$, (ii) a desired number of clusters K, and (iii)an *objective function* that evaluates the quality of OBJECTIVE FUNCTION a clustering, we want to compute an assignment $\gamma : D \rightarrow \{1, \ldots, K\}$ that minimizes (or, in othercases, maximizes) the objective function. In most cases, we also demand thatγ is subjective, i.e., that none of the K clusters is empty.

The objective function is often defined in terms of similarity or distancebetween documents. The objective in K-means clusteringis to minimize the average distance between documents and their centroidsor, equivalently, to maximize the similarity between documents andtheir centroids. For documents, the type of similarity we want is usually topic similarityor high values on the same dimensions in the vector space model. For example, documents

about China have high values on dimensions like Chinese,Beijing, and Mao whereas documents about the UK tend to have high valuesfor London, Britain and Queen. We approximate topic similarity with cosinesimilarity or Euclidean distance in vector space .If we intend tocapture similarity of a type other than topic, for example, similarity of language, then a different representation may be appropriate. When computingtopic similarity, stop words can be safely ignored, but they are importantcues for separating clusters of English (in which theoccurs frequently and lainfrequently) and French documents (in which the occurs infrequently and lafrequently).Thatis, the only background knowledge availableis the number of clusters we want to groupthe documents in (K), which usually coincideswith the expected number of thematiccategories contained in the corpus. Vector Space Model(VSM) represents a document as a vector of terms (orphrases) in which each dimension corresponds to a term(or a phrase). An entry of a vector is non-zero if thecorresponding term (or phrase) occurs in the document. Asignificant progress has been made with vector space modelin many applications. However, it has limitations due toits oversimplification of a document to a term vector. Forexample, long documents usually contain richer informationthan short ones, but long documents represented withhigh-dimensional vectors result in calculations of documentsimilarities that are susceptible to noise. Also one cannotexplicitly represent topics in vector space model[2].Our method deals with the problem by stemming the words in the document and reducing noise to a large extent in effect.

2 Related Works

The hybrid approach described in[3] combines similarity measures, defined by a content-based distance, and a classical distribution-based measure together with a behavioral analysis of the style features of the compared documents.The authors mention that the novel aspect of the method described here is the use of a document-distance that takes into account both a conventional content-based similarity metric and a behavioral similarity criterion.The Vector space model was chosen for information extraction. Given a collection of documents D, the vector space model represents each document D as a vector of real-valued weight terms $\mathbf{v}=\{wj;\ j=1,..,nT\}$. Each component of the nT-dimensional vectoris a non-negative term weight, wj, characterizing the j^{th} term and denoting the relevance of the term itself within the document D.Therefore, the k-th element of the vector $\mathbf{v}'(Du)$ is defined as:

$$v'_{k,u}=\frac{tf_{k,u}}{\sum_{l=1}^{n_T} tf_{l,u}}$$

where $tf_{k,u}$is the frequency of the k-th term in document Du.Thus, $\mathbf{v}'$ represents a document by a classical vector model,and uses term frequencies to set the weights associated toeach element. Gaussian distribution was assumed to identify the spatial probability density of a term$t$ in a set of documents Du. It can be understood that while the Vector space model is useful for calculating frequency-based distance, the spatial probability density function (pdf) aids in calculating the behavioral distance.While Minkowski distance was used to calculate the frequency-based distance, Euclidean distance was used as the behavioral distance measure.Both terms contribute to the computation of theeventual distance value was calculated as the weighted sum of the two distances.

The kernel-based version of the k-means algorithm,Used replicates thebasic partitioning schema in the Hilbert space, wherethe centroid positions, Ψ, are given by the averages of the mapping images, Φu:

Finally the distances from the mapped image to the cluster centroids are calculated to identify the closest prototype to theimage of each input pattern, and assign sample membershipsaccordingly. Experimental results for purity on Newsgroup 20 using the above clustering method indicate a clear increase in the overall purity with increasing number of clusters.

3.Hybrid Based Distance Clustering

Although the above clustering Framework in [3] yields superior results in the case of increasing number of clusters for a particular corpus, however, the effect of increase in Corpus with fixed number of Clusters is left to be desired. Moreover, whether only keywords are considered or both keywords and phrases is not very clearly specified.While the previous work emphasizes on improving the purity by combining content-based similarity metric and a behavioral similarity criterion, and using combined distance measures for the final clustering process. The emphasis here has been on using effective distribution and distance measures for improving the purity. Our present work focuses on improving the purity by combining keyword based clustering and Phrase index based clustering. In other words, our

focus has been more on improving the content-based similarity and achieving results almost near to the ones obtained in the previous work.

Our clustering framework is summarized in the following steps:

Step 1: Phrase extraction - which is a text mining task, extracts highly relevant phrases from documents. A keyphrase is "a sequence of one or more words that is considered highly relevant", while a keyword is "a single word that is highly relevant." An arbitrary combination of keywords does not necessarily constitute a keyphrase; neither do the constituents of a keyphrase necessarily represent individual keywords[4].

Step 2: Phrase indexing - which assigns indices to the extracted Phrases & calculates the frequency of occurrence of Phrases within the document.

Step 3: Term Filtering - The removal of stopwords is the most common term filtering technique used. There are standard stopword lists available but in most of the applications these are modified depending on the quality of the dataset. Our method uses removal of terms with low document frequencies. This is done to improve the speed and memory consumption of the application.

Step 4: Stemming - Stemming is the process of reducing words to their stem or root form. For example 'cook', 'cooking', 'cooked' are all forms of the same word used in different constraint but for measuring similarity these should be considered same.

Step 5: Index Substitution - Replacing the phrases and the stemmed words with the respective indices.

Step 6: Representation of documents in Vector Space Model - Many issues specific to documents are discussed more fully in information retrieval texts. We briefly review a few essential topics to provide a sufficient background for understanding document clustering.

For our clustering algorithms documents are represented using the vector space model. In this model each document is considered to be a vector in the term space(set of document words i,e; vocabulary). In the simplest form each document is represented by *tf* vector,
$d_{tf} = (tf_1, tf_2, \ldots tf_n)$,

where tf_i is the frequency of i^{th} term in the document d. in addition we use the version of this model based on its inverse document frequency (IDF) in the document collection. Finally in order to account for documents of different lengths, each document vector is normalized so that it is of unit length.

Step 7: Estimation of Hybrid Distance - our approach considers the computation of frequency based distance d(*f*) and the behavioral distance ,d(b) given by

$$d(f)(D_x, D_y) = \left[\sum_{i=1}^{n} |tf_{x,i} - tf_{y,i}|^{\lambda} \right]^{1/\lambda} \quad (1)$$

$$d(b)(D_x, D_y) = \left[\sum_{i=1}^{n} |D_{x,i} - D_{y,i}|^{2} \right]^{1/2} \quad (2)$$

Where n represents the number of terms. The present approach adopts the value λ=1, which actually implements a Manhattan distance metric.
Terms (1) and (2) leads to the computation of the hybrid distance given by

$$h(Dx, Dy) = \beta.d(f)(Dx, Dy)+(1-\beta).d(b)(Dx, Dy) \quad (3)$$

where the coefficient β lies in [0,1].

Step 8: Applying k-means algorithm for clustering - For k-means clustering, the cosine measure is used to compute which document centroid is closest to the given document.

Step 9: Purity evaluation - since the purpose of our clustering is to classify clusters of texts rather than single texts the *purity* of each cluster is an appealing measure:

$$\rho_i = \max_j \{p_{ij}\}.$$

We may use the weighted average purity overall clusters as a measure of quality of the whole clustering defined as

$$\rho = \sum_i (n_i/n)\rho_i = n_{max}/n$$

where n is the total number of documents and n_{max} is the number of documents in the entire set that are part of a cluster where the number of documents from their classes is greater than the number of documents from other classes[6].

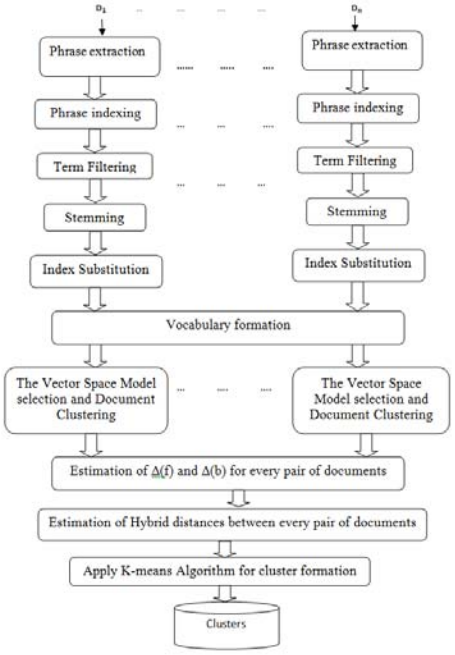

Figure 1: Framework for Hybrid Distance Based Clustering

4. Experimental Results

For the purpose of experimentation, we had chosen the Newsgroup 20 dataset which is a very standard and popular dataset used for evaluation of many text applications, data mining methods, machine learning methods, etc. It has 18,828 unique documents organized in 20 categories. The Purity parameter was chosen to evaluate the performance of our clustering framework. Let N_k denote the number of elements lying in a cluster C_k and let N_{mk} be the number of elements of the class I_{min} the cluster C_k. Then, the purity *purity*(k) of the cluster C_k is defined as follows:

$$purityof\ k^{th}\ cluster = \frac{\#\ elementsinmajorityclass}{Sizeofk^{th}cluster}$$

The overall purity,'O' of the clustering results is defined as follows:

$$O = \sum \left(\frac{\#\ elementsinkthcluster}{\#\ \text{documents in the corpus}} \right) purity(k)$$

The experiments involved two sets, one with varying corpora that includes increasing number of classes while maintaining a constant number of clusters. The other set involved in a fixed corpora with increasing number of clusters.The overall purity of clusters was measured in both the cases.

Results obtained for combination of phrases and keywords applied on Newsgroup-20 dataset for varying number of classes with fixed number of clusters is shown in table 1 and figure 2.

No of Clusters	No of Classes	Min Purity	Max Purity	Overall Purity	Smallest Cluster
40	5	0.004028	1	0.142811	6
40	10	0.004028	1	0.168091	6
40	15	0.004028	1	0.18344	6
40	20	0.007479	1	0.29865	6

Table 1: Varying number of **classes** with fixed number of **clusters**

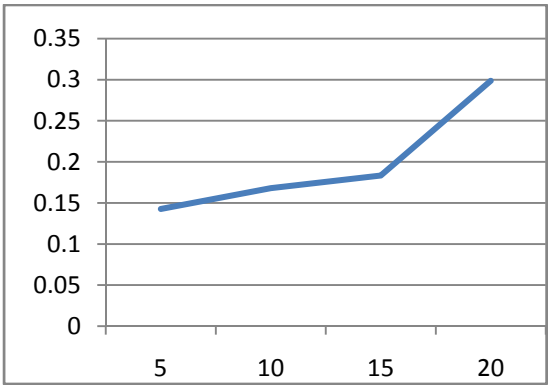

Figure 2: Varying number of **classes** with fixed number of **clusters**

No of Clusters	No of Classes	Min Purity	Max Purity	Overall Purity	Smallest Cluster
10	20	0.0621141	1	0.19796	31
20	20	0.06779181	1	0.219916	16
40	20	0.07478633	1	0.29865	6
80	20	0.07416564	1	0.350485	5
100	20	0.07375538	1	0.356871	5

Table 2: Varying number of **clusters** with fixed number of **classes**

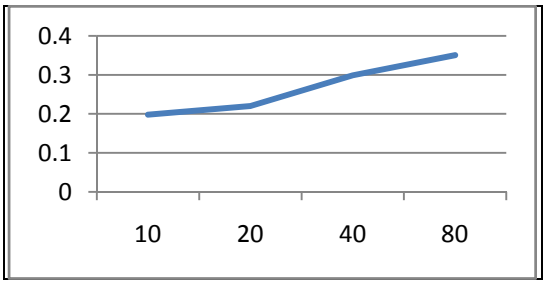

Figure 3: Varying number of **clusters** with fixed number of **classes**

The results have been compared with those obtained in the previous work[3](on *DN2* with α=0.3).For the purpose of comparison, we have included all the documents associated with the same five categories considered previously. Corpora consisting of comp.graphics, comp.os.ms-windows, rec.autos, and sci.electronics classes from newsgroup dataset. The result obtained in previous work[3] are shown in table 3 and the result obtained using method proposed in this paper are tabulated in table 5(for clustering using keywords only) and table 4(for clustering using keywords and phrases). Also, results obtained are pictorially depicted in figures 5 and 6 respectively.

Number of clusters	Overall purity	*pur*(*k*) minimum	*pur*(*k*) maximum	Smallest cluster
20	0.627895	0.303704	1	20
40	0.679817	0.298701	1	14
60	0.691016	0.295775	1	4
80	0.657928	0.265306	1	5
100	0.695597	0.349515	1	5

Table 3: Clustering performances obtained on *DN2* with α=0.3

No. Of cluster	**Overall purity**	**Min p**	**Max p**	**Smallest Cluster**
20	0.660299	0.335	0.877	11
40	0.702615	0.303	1	2
60	0.77461	0.2949	1	3
80	0.821334	0.296296	1	1
100	0.850342	0.28	1	1

Table 4: Keywords and phrases in DN2 dataset

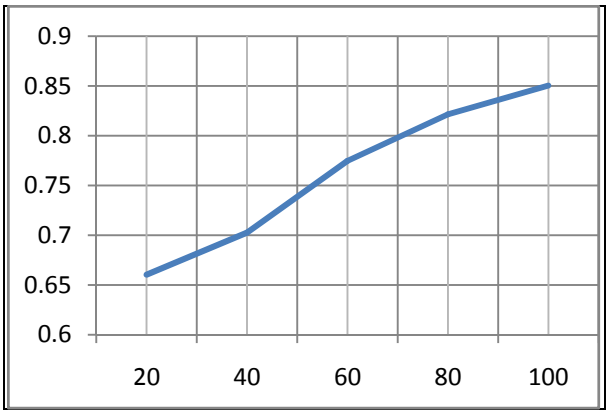

Figure 5: Keywords and phrases in DN2 dataset

No. Of cluster	Overall purity	Min p	Max p	Smallest
20	0.660594	0.335079	0.873016	11
40	0.733177	0.30303	1	3
60	0.773729	0.259259	1	3
80	0.82104	0.297945	1	1
100	0.862413	0.29927	1	1

Table 5: Keywords only in DN2 dataset

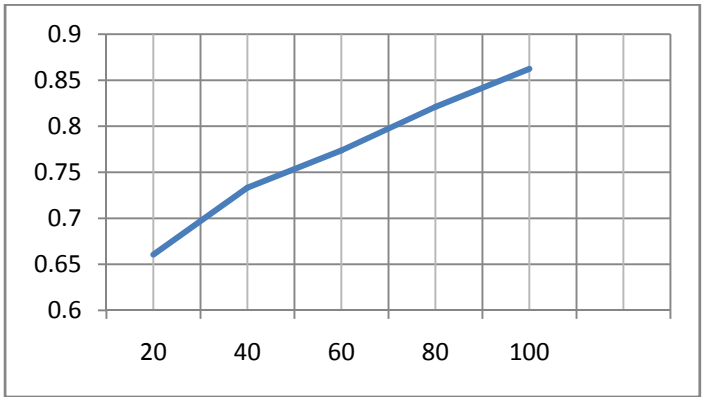

Figure 6: Keywords only in DN2 dataset

Results indicate a steady increase in the overall purity while the minimum purity is comparably close to the text clustering framework proposed previously. Also, the smallest cluster size was low in all cases for our proposed method which

makes it even more ideal in cases where the cluster space is restricted.

5. Conclusions

Document clustering is being studied from many decades but still it is far from a trivial and solved problem.
The challenges are:
1. Selecting appropriate features of the documents that should be used for clustering.
2. Selecting an appropriate similarity measure between documents.
3. Selecting an appropriate clustering method utilizing the above similarity measure.
4. Implementing the clustering algorithm in an efficient way that makes it feasible in terms of required memory and CPU resources.
5. Finding ways of assessing the quality of the performed clustering.

In our work, we have proposed a **"Hybrid Distance Based Document Clustering With Keyword And Phrase Indexing"** that uses an improved indexing and substitution method for document phrases. Though lots of methods have previously been proposed that consider either Keyword or Phrases for measuring content-based similarity, very few methods consider both. Although our method uses the conventional K-means for distance measure, yet it delivers a superior performance in terms of purity owing to the mechanism employed for measuring the content-similarity. Future improvements could be combining the behavioral distance methods together with our improved content-based similarity measure to further improve the performance of clustering.

References

[1] LoulwahAlSumait and Carlotta Domeniconi: "Local Semantic Kernels for Text Document Clustering"
[2] Xufei Wang, Jiliang Tang and Huan Liu: "Document Clustering via Matrix Representation", 2011.
[3] Sergio Decherchi, Paolo Gastaldo, Judith Redi and Rodolfo Zunino: ":**"**A Text Clustering Framework for Information Retrieval**"**, Journal of Information Assurance and Security, Special Issue CISIS2008,2009.
[4] Khaled M. Hammouda, Diego N. Matute, and Mohamed S. Kamel: "CorePhrase: Keyphrase Extraction for Document Clustering"
[5] Michael Steinbach, George Karypis and VipinKumar: "A Comparison of Document Clustering Techniques"
[6] Magnus Rosell, ViggoKann and Jan-Eric Litton: "Comparing Comparisons: Document Clustering Evaluation Using Two Manual Classifications"
[7] Data Mining:Concepts and Techniques by Jiawei Han, Mic