International Journal of Computer Applications in Engineering Sciences
[VOL I, ISSUE IV, DECEMBER 2011] [ISSN: 2231-4946]

Performance Analysis of Standard k-Means Clustering Algorithm on Clustering TMG format Document Data

*P. Perumal [1], R. Nedunchezhian [2]

[1] *Department of CSE, Sri Ramakrishna Engineering College, Coimbator, India*
[2] *Department of IT, Sri Ramakrishna Engineering College, Coimbator, India*

[1]perumalsrec@gmail.com
[2]rajuchezhian@yahoo.co.uk

***Abstract*- Document clustering is useful in many information retrieval operations such as document browsing, organization and viewing of retrieval results, generation of Yahoo-like hierarchies of documents, etc. The general goal of clustering is to group data elements such that the intra-group similarities are high and the inter-group similarities are low. Generative models based on the multivariate Bernoulli and multinomial distributions have been widely used for text classification. In this work, we explore the k-means clustering algorithm for document clustering problem. The proposed work implements the standard k-mean clustering algorithm and tests it with TMG format document data and L2-normalized document data. The results of the k-means clustering algorithm are compared with von Mises-Fisher model-based clustering (vMF-based k-means) algorithm.**

***Key Words*- Text Clustering, Classification, Document Clustering, Model Based Clustering, Term Document Matrix, Text to Matrix Generator (TMG), k-means, Mises-Fisher Clustering**

I. INTRODUCTION

Clustering

Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects. A cluster is a collection of data objects that are similar to one another with the same cluster and are dissimilar to the objects in other clusters. A cluster of data objects can be treated collectively as one group in many applications. Clustering is a form of learning by observation rather than learning by examples. Cluster analysis is an important human activity in which we indulge since childhood when we learn to distinguish between animals and plants etc by continuously improving subconscious clustering schemes. It is widely used in numerous applications, including pattern recognition, data analysis, image processing, and market research etc.,[8][17][18].

Clustering is a very important application area but widely interdisciplinary in nature, that makes it very difficult to define its scope. It is used in several research communities to describe methods for grouping of unlabeled data. Now, these communities have different terminologies and assumptions for the components of the clustering process and the contexts in which clustering are used. Cluster analysis has been studied extensively for years, focusing mainly on distance-based cluster analysis. Many clustering tools were made based on k-means, k-medoids, and some of the methods were incorporated in many statistical analysis software packages [17] [18]. The major clustering steps are preprocessing and feature selection, similarity measure, clustering algorithms, result validation, and result interpretation.

In this work, the suitability of k-means clustering algorithm for document clustering application has been studied and its performance along with another probabilistic method von Mises-Fisher model-based clustering is compared.

Document Clustering

The document clustering is the core topic in the information retrieval field. It uses unsupervised algorithms to cluster large amount of web page into several groups. Let's take an example to illustrate why document clustering is necessary. Everyone has experienced Google search for information from the Internet. In response to a query of a web client, Google will send back tons of web pages. Although they are listed by the order of its importance, users still sometimes have to browse hundreds of web page to find what they want. If we can group the pages into groups, users can skip the group they are not interested in. They will not have to browse too many pages before reaching their targets. This will help users to execute their queries efficiently. However the problem is How to group

pages? The answer is using document clustering [16] [17].

Key requirements for document clustering are:

- How to present a document in the mathematical model
- What document clustering algorithms to use
- How to refine to the clustering algorithm
- How to choose an appropriate topic to present the clusters.
- How to evaluate the algorithms and compare the resulting clusters.
- How to apply to real world document clustering application.

Document clustering has become an increasingly important technique for unsupervised document organization, automatic topic extraction, and fast information retrieval or filtering [16][17].

II. PROBABILISTIC CLUSTERING MODELS UNDER EVALUATION

Model-based Partitional Clustering

The model-based k-means (mk-means) algorithm is a generalization of the standard k-means algorithm, with the cluster centroid vectors being replaced by a probabilistic model. Let $X = \{x_1, \ldots, x_N\}$ be the set of data object and $\wedge = \{\lambda_1, \ldots, \lambda_K\}$ the set cluster models. The mk – means algorithm locally maximizes the log – likelihood objective function

$$\log P(X \mid \wedge) = \sum_{x \varepsilon X} \log p(x \mid \lambda_{y(x)}),$$

where y(x) = $\arg\max_y \log p(x \mid \lambda_y)$ is the cluster identity of object x [18].

The traditional vector space representation is used for text documents, i.e., each document is represented as a high dimensional vector of "word" counts in the document. The dimensionality equals the number of words in the vocabulary used.

von Mises-Fisher Model

The von Mises-Fisher distribution is the analogue of the Gaussian distribution for directional data, in the sense that it is the unique distribution of L2-normalized data that maximizes the entropy, given the first and second moments of the distribution (Mardia, 1975). It has recently been shown that the spherical k-means algorithm that uses the cosine similarity metric (to measure the closeness of a data point to its cluster's centroid) can be derived from a generative model based on the vMF distribution under certain restrictive conditions (Banerjee & Ghosh, 2002; Banerjee et al., 2003). The vMF distribution for cluster j can be written as

$$P(d_i \mid \lambda_j) = \frac{1}{Z(k_j)} \exp(kj \frac{d_i^T \mu j}{\| \mu_j \|}),$$

where d_i is a normalized document vector and the Bessel function $Z(k_j)$ is a normalization term. The parameter k measures the directional variance and the higher it is, the more peaked the distribution is. For the vMF-based k-means algorithm, we assume k is the same for all clusters, i.e., $k_j = k, \forall_j$. This results in the spherical k-means (Dhillon & Modha, 2001; Dhillon et al., 2001) [18]. The model estimation in this case simply amounts to $\mu_i = \frac{1}{n_j} \sum_{i:y_i=j} d_i$, where n_j is the number of documents in cluster j.

A. The Standard K-means Algorithm

In this method, the standard k-means clustering algorithm is applied on the original document term frequency data set (tgm format)

```
Inputs: X= {x1,...,xk}     (the document vectors to be
            clustered)
        n  (the number of clusters)
Outputs: C= {c1… cn}       (the Cluster Centroids)
         m: X → {1...n} (the cluster membership)
Procedure k-means {
        Randomly initialize C
        For each xi∈ X {
                m (xi)=argmin distance(xi, cj)
                        j∈{1..n}
        }
       While m has changed {
            For each i ∈ {1...n} {
                    Recomputed Ci as the centroid of
                      {x|m(x) = i}
             }
              For each xi∈ X {
                    m (xI) = argmin distance (xi, cj)
                            j∈{1..n}
              }
        }
}
```

Metrics Considered for Evaluation

Validating clustering algorithms and comparing performance of different algorithms is complex, because it is difficult to find an objective measure of quality of clusters. In order to compare results against external criteria, a measure of agreement is needed. Since we assume that each record is assigned to only

one class in the external criterion and to only one cluster, measures of agreement between two partitions can be used.

An important aspect of cluster analysis is the evaluation of clustering results. Halkidi, M. et al. (2001) made a comprehensive review of clustering validity measures available in the literature and classified them into three categories. In this section we briefly review the commonly used document clustering evaluation measures and the evaluation of search results clustering in the literature.
The first is the external evaluation method, which evaluates the results of clustering algorithm based on a pre-classified document set. There are several ways of comparing the clusters with the pre-defined classes: Rand Index, purity and mutual information.

Rand Index

The Rand index or Rand measure is a commonly used technique for measure of such similarity between two data clusters.
Given a set of n objects S = {O1... On} and two data clusters of S which we want to compare: X = {x1... xR} and Y = {y1... yS} where the different partitions of X and Y are disjoint and their union is equal to S; we can compute the following values:

- a is the number of elements in S that are in the same partition in X and in the same partition in Y,
- b is the number of elements in S that are not in the same partition in X and not in the same partition in Y,
- c is the number of elements in S that are in the same partition in X and not in the same partition in Y,
- d is the number of elements in S that are not in the same partition in X but are in the same partition in Y.

Intuitively, one can think of a + b as the number of agreements between X and Y and c + d as the number of disagreements between X and Y. The rand index, R, then becomes,

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}}$$

The rand index has a value between 0 and 1 with 0 indicating that the two data clusters do not agree on any pair of points and 1 indicating that the data clusters are exactly the same.

Purity

Purity can be computed by assigning class labels for each cluster by majority voting, then, for a single cluster calculating the ratio of the correctly labeled documents to the total number of documents in the cluster. Let there be k clusters (the k in k-means) of the dataset D and the size of cluster Cj be |Cj |. Let |Cj |class=i denote number of items of class i assigned to cluster j. Purity of this cluster is given by [23]

$$purity(C_j) = \frac{1}{|C_j|} \max_i (|C_j|_{class=i})$$

The overall purity of a clustering solution could be expressed as a weighted sum of individual cluster purities.

$$purity = \sum_{j=1}^{k} \frac{|C_j|}{|\mathcal{D}|} purity(C_j)$$

In general, larger value of purity means that better solution.

Mutual Information

We use the mutual information between an element (document) and its features (terms).In our algorithm, for each element *e*, construct a frequency count vector *C*(*e*) = (c_{e1}, c_{e2}, …, c_{em}), where *m* is the total number of features and c_{ef} is the frequency count of feature *f* occurring in element *e*. In document clustering, *e* is a document and c_{ef} is the term frequency of *f* in *e*. We construct a mutual information vector *MI*(*e*) = (mi_{e1}, mi_{e2}, …, mi_{em}), where mi_{ef} is the mutual information between element *e* and feature *f*, which is defined as[24]:

$$mi_{ef} = \log \frac{\frac{c_{ef}}{N}}{\frac{\sum_i c_{if}}{N} \times \frac{\sum_j c_{ej}}{N}}$$

where $N = \sum_i \sum_j c_{ij}$ is the total frequency count of all features of all elements.

We compute the similarity between two elements e_i and e_j using the *cosine coefficient* **Error! Reference source not found.**of their mutual information vectors [24]:

$$sim(e_i, e_j) = \frac{\sum_f mi_{e_i f} \times mi_{e_j f}}{\sqrt{\sum_f mi_{e_i f}^{\,2} \times \sum_f mi_{e_j f}^{\,2}}}$$

B. The Standard K-means Algorithm with of L2-normalized data

In this method, instead of using the original document term frequency data set (tgm format), the L2 normalized data of the term frequency data set is used for k-means clustering.

Let X= $\{x_1,\ldots,x_k\}$ be the document vectors to be clustered, then the l^2-norm of the vector is given by

$$|\mathbf{x}| = \sqrt{x_1^2 + x_2^2 + x_3^2}\ .$$

The l^2-norm is also known as the Euclidean norm.

III. Implementation And Evaluation

To evaluate the algorithms, a suitable and standard data set is needed. We decided to use some of the same datasets which were originally used in a previous reference work [16] [17]. The datasets were originally from TREC collections (http://trec.nist.gov). Datasets tr11, tr23, tr41, and tr45 were originally derived from TREC-5, TREC-6, and TREC-7 collections (NIST Text REtrieval Conferences - TREC). In this paper, we have used TMG format of these datasets which is sited in different previous works. We selected these data sets for evaluation because of their standard.

Performance in Terms of CPU time

In the following table we present the outputs of time study made on a Windows XP laptop equipped with Intel core 2 duo CPU at 2GHz and 2GB RAM. The Matlab implementations of the algorithms were used for evaluation. As shown in the following tables and graphs, the performance in terms of CPU time was very high in k means clustering of tgm data as well as L2-normalized data. The performance of the probabilistic model-von Mises-Fisher based k-means clustering was very good since the distance computation part is entirely different than the normal k-means clustering method.

Table 1: Accuracy in Terms of CPU time

Data Set Used and its size (rows x Columns)	Time Taken for Clustering (Average of Three runs)		
	von Mises-Fisher based k-means	Std k-means	k-means with L2-normalized data
Tr11 414 x 6424	0.4530	1.078	1.219
Tr12 313 x 5799	0.2500	1.125	0.828
Tr23 204 x 5831	0.1720	0.469	0.531
Tr31 927 x 10127	1.0470	2.109	2.094
Tr41 690 x 8261	0.3593	2.39	1.906
Tr45.mat 690 x 8261	0.4210	2.859	1.75
La2.mat 3075 x 31472	1.2920	4.391	4.5
La12.mat 6279 x 31472	2.9060	6.422	8.718
Avg	0.862538	2.605375	2.69325

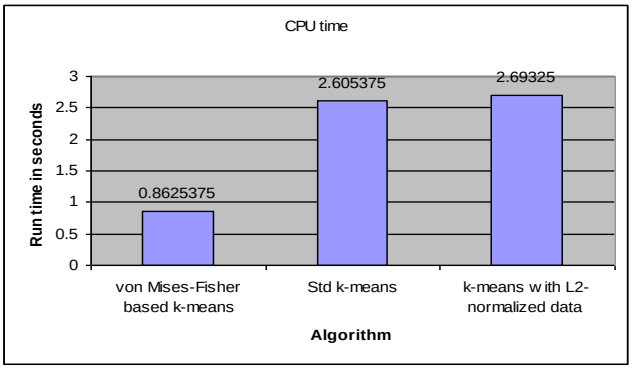


Figure 1: Performance in terms of CPU time

The performance of Clustering with Different Datasets

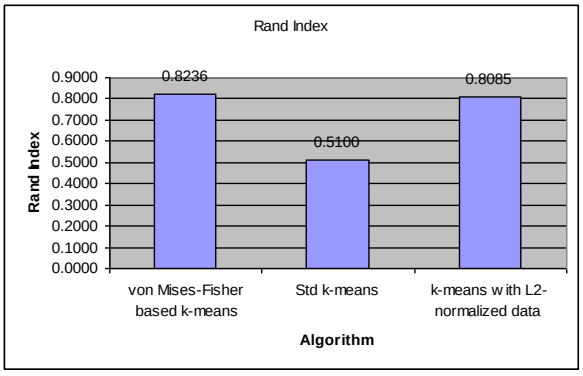
Clustering Accuracy in Terms of Rand Index

Table 2: Accuracy in Terms of Rand Index with Different Data Sets

Data Set Used and its size (rows x Columns)	Clustering Accuracy in Terms of Rand Index (Average of Three runs)		
	von Mises-Fisher based k-means	Std k-means	k-means with L2-normalized data
Tr11 414 x 6424	0.8387	0.4148	0.8484
Tr12 313 x 5799	0.8308	0.3302	0.8314
Tr23 204 x 5831	0.6932	0.4628	0.6924
Tr31 927 x 10127	0.8184	0.4118	0.7859
Tr41 690 x 8261	0.8847	0.6851	0.8743
Tr45.mat 690 x 8261	0.8821	0.6706	0.8741
La2.mat 3075 x 31472	0.8161	0.5033	0.7742

La12.mat 6279 x 31472	0.8248	0.6013	0.7869
Avg	0.8236	0.5100	0.8084

In our previous work [17] [18], we showed that the Rand index is the most suitable quality metric. Our previous result clearly signifies that fact. As shown in the above table and the following figure, the performance of k-means clustering was very poor if we apply it directly on the TGM format data. If we apply the same algorithm on the L2-normalized data, then it achieves performance almost equal to that of the probabilistic model - von Mises-Fisher based k-means clustering.



.

Figure 2: Rand Index - Comparison Graph

Clustering Accuracy in Terms of Mutual Information

As shown in the following tables and graphs, the performance in terms of mutual information measure was good in the case of k means clustering of L2-normalized data.

Table 3: ACCURACY IN TERMS OF MUTUAL INFORMATION MEASURE WITH DIFFERENT DATA SETS

Data Set Used and its size (rows x Columns)	Clustering Accuracy in Terms of Mutual Information (Average of Three runs)		
	von Mises-Fisher based k-means	Std k-means	k-means with L2-normalized data
Tr11 414 x 6424	0.2150	0.1183	0.5727
Tr12 313 x 5799	0.2133	0.0876	0.4038
Tr23 204 x 5831	0.2015	0.1503	0.3199
Tr31 927 x 10127	0.2102	0.1002	0.5163
Tr41 690 x 8261	0.2346	0.3136	0.6365
Tr45.mat 690 x 8261	0.4825	0.2675	0.6105
La2.mat 3075 x 31472	0.4841	0.1177	0.3717
La12.mat 6279 x 31472	0.5008	0.1393	0.4741
Avg	0.3177	0.1618	0.4882

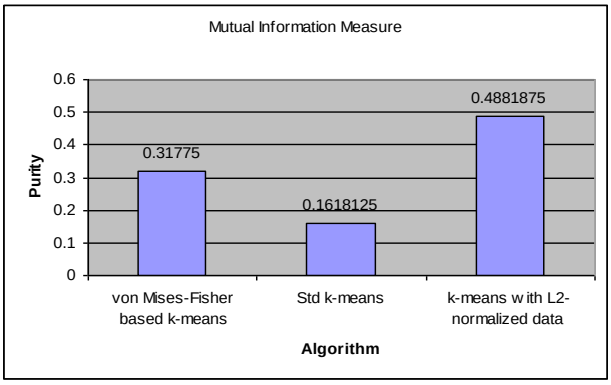


Figure 3: Mutual Information - Comparison Graph

As shown in the following table and graph, the performance in terms of purity measure was good in the case of k means clustering of L2-normalized data.

Table 4: ACCURACY IN TERMS OF PURITY WITH DIFFERENT DATA SETS

Data Set Used and its size (rows x Columns)	Clustering Accuracy in Terms of Purity Measure (Average of Three runs)		
	von Mises-Fisher based k-means	Std k-means	k-means with L2-normalized data
Tr11 414 x 6424	0.2150	0.6124	0.7352
Tr12 313 x 5799	0.2133	0.5216	0.4716
Tr23 204 x 5831	0.2015	0.7286	0.6783
Tr31 927 x 10127	0.2102	0.5789	0.7553
Tr41 690 x 8261	0.2346	0.6837	0.7942
Tr45.mat 690 x 8261	0.6804	0.6166	0.7495
La2.mat 3075 x 31472	0.7161	0.6617	0.6055
La12.mat 6279 x 31472	0.7051	0.6106	0.7321
Avg	0.3970	0.6268	0.6902

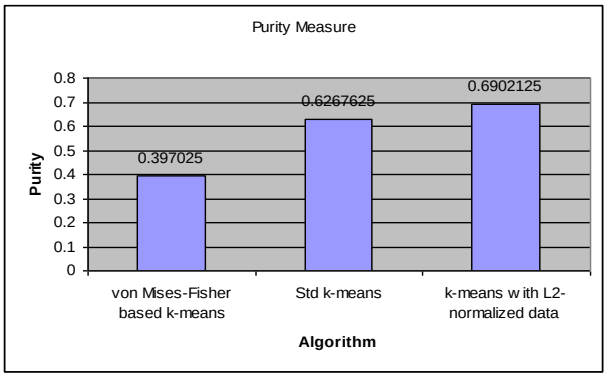


Figure 4: Purity - Comparison Graph

IV. CONCLUSION AND SCOPE FOR FURTHER ENHANCEMENTS

In this paper, we have evaluated three clustering algorithm towards their performance in document clustering application. Rand Index and CPU time are used as the main metrics during evaluating these algorithms. The arrived results were more significant and more comparable. It was realized that the performance of the normal k-means clustering with the L2-normalized data is little bit lesser or almost equal to that of von Mises-Fisher based clustering algorithm. But the time consumed for normal k-means clustering is significantly higher than that of Mises-Fisher based method.

Through comparison of performance of these three algorithms in terms of Rand Index and Mutual Information Measure (which is used in previous work [16] [17]), it was obvious that the standard k-mean clustering can not directly classify the document data which is in the form of term frequency count. The suitable preprocessing enhanced the performance of the k-means clustering and achieves the better results like von Mises-Fisher based k-means algorithm. But, the performance of standard k-mean clustering algorithm was very poor than the von Mises-Fisher based k-means algorithm. This is due to the higher dimensionality of the document data under consideration.

Future works may address the possibilities of improving the standard k-means approach for better classification accuracy and improved performance in terms of CPU time. We observed that the Term-Document Matrix which is generally used to represent the documents as vectors itself affects the performance of the k-means clustering algorithm. In experience it was observed that there will be large difference in magnitude of individual attribute of data in the Term-Document Matrix format of representation. This very basic nature of this data, leads to poor accuracy in clustering due to the problems in finding distance between records using individual attributes in the case of normal k-means clustering. We are exploring data transformation techniques and some minor changes in distance calculations to enhance the performance of text clustering algorithm. Future work will address these problems and design a more improved document clustering algorithm.

V. ACKNOWLEDGEMENT

We thank our Director, Principal and the management of Sri Ramakrishna Engineering College for providing lab facility to implement this work.

REFERENCES

[1] Aas, K., Eikvil, L., "Text Categorisation: A survey," Technical report, Norwegian Computing Center, P.B. 114 Blindren, N-0314 Oslo, Norway, June 1999.

[2] Can, Fazli; Ozkarahan, Esen A., 1985. "Similarity and Stability Analysis of the Two Partitioning Type Clustering Algorithms", Journal of the American Society for Information Science, 36(1):3-14, 1985.

[3] Fang, Y.C., Parthasarathy, S and Schwartz, F., "Using clustering to Boost Text Classification," In:Workshop on Text mining, TextDM 2001.

[4] Florian Beil, Martin Ester, "Frequent Term-Based Text Clustering," workshop on KDD 2002.

[5] Hartigan J.A., Clustering Algorithms. John Wiley and Sons, NY, 1975.

[6] Jain, A.K. and Dubes, R.C. Algorithms for clustering Data, Prentice-Hall advanced reference series, Prentice- Hall Inc., Upper Saddle River, NJ, 1988.

[7] Jain, A.K., Murty, M.N., and Flynn, P.J., "Data Clustering – Survey", ACM Computing Surveys, Vol. 31, No.3, pp. 264 – 323, 1999.

[8] Han, J. W. and Kamber, M. Data Mining: Concepts and Techniques, 2nd edition, Morgan Kaufmann Publishers, March 2006.

[9] J.L.Neto, A.D.Santos, C.A.A. Kaestner, and A.A. Freitas, Document Clustering and Text Summarization," 4th International Conference on Practical Applications of Knowledge Discovery and Data Ming, London, 2000.

[10] Maarek. S, Ronald Fagin, Israel Z. Ben-Shaul, Dan Pelleg, "Ephemeral Document Clustering for Web Applications," IBM Research Report RJ 10186, April, 2000.

[11] McCallum and K. Nigam, "A comparison of event models for naive Bayes text Classification,"AAAI Workshop on Learning for Text Categorization, 1998.

[12] Mei-Ling Shyu, Shu-Ching Chen, Et Al., "Affinity-Based Probabilistic Reasoning and Document Clustering on the WWW," COMPSAC, pp.149-154, 2000.

[13] Michael Steinbach, George Karypis, Vipin Kumar, "A Comparison of Document Clustering Techniques," Proc. TextMining Workshop, KDD 2000, 2000.

[14] Radecki, and Tadeusz, "Probabilistic Methods for Ranking Output Documents in Conventional Boolean Retrieval Systems," Inf. Process. Manage. 24(3): 281- 302, 1998.

[15] Ravichandra Rao, 2003. "Data Mining and Clustering Techniques", DRTC Annual workshop on Semantic Web, Paper K: 1-12.

[16] P. Perumal, R. Nedunchezhian, "Performance Evaluation of Three Model-Based Documents Clustering Algorithms," European journal of Scientific Research, Vol.52 No.4 (2011), pp.618-628.

[17] P. Perumal, R. Nedunchezhian, "Improving the Performance of Multivariate Bernoulli Model based Documents Clustering

Algorithms using Transformation Techniques," Journal of Computer Science, 7 (5): 762-769, 2011.
[18] S. Zhong, S and J. Ghosh, J, "A comparative study of generative models for document clustering", SDM Workshop on Clustering High Dimensional Data and Its Applications, May 2003.
[19] S.M. Rüger, S.E. Gauch, "Feature Reduction for Document Clustering and Classification," Technical report, Computing Department, Imperial College London, UK, 2000.
[20] Sholom Weiss, Brian White, Chid Apte , "Lightweight Document Clustering," IBM Research Report RC-21684, 2000
[21] Tryon, R.C., and Bailey, D.e. Cluster Analysis, McGraw-Hill, New York, 1970.
[22] Turenne, Nicolas; Roussillon, Francois, "Evaluation of Four Clustering Methods Used in Text Mining," Proceedings of ECML Workshop on TextMining, 1998.
[23] Xiaoxia Wang and Max Bramer, "Exploring web search results clustering," Research and development in Intelligent systems XXIII: proceedings of AI -2006, the 26th International Conference on Innovative techniques and applications of AI, pp. 393-397, 2006.
[24] Patrick Pantel, and Dekang Lin., "Efficiently Clustering Documents with Committees," In: PRICAI, Vol. 2417 Springer, p. 424-433, 2002.