Machine Learning, 55, 311–331, 2004

Empirical and Theoretical Comparisons of Selected Criterion Functions for Document Clustering*

YING ZHAO yzhao@cs.umn.edu
GEORGE KARYPIS karypis@cs.umn.edu
University of Minnesota, Department of Computer Science Minneapolis, MN 55455, USA

Editor: Douglas Fisher

Abstract. This paper evaluates the performance of different criterion functions in the context of partitional clustering algorithms for document datasets. Our study involves a total of seven different criterion functions, three of which are introduced in this paper and four that have been proposed in the past. We present a comprehensive experimental evaluation involving 15 different datasets, as well as an analysis of the characteristics of the various criterion functions and their effect on the clusters they produce. Our experimental results show that there are a set of criterion functions that consistently outperform the rest, and that some of the newly proposed criterion functions lead to the best overall results. Our theoretical analysis shows that the relative performance of the criterion functions depends on (i) the degree to which they can correctly operate when the clusters are of different tightness, and (ii) the degree to which they can lead to reasonably balanced clusters.

Keywords: partitional clustering, criterion function, data mining, information retrieval

1. Introduction

The topic of clustering has been extensively studied in many scientific disciplines and a variety of different algorithms have been developed (MacQueen, 1967; King, 1967; Zahn, 1971; Sneath & Sokal, 1973; Dempster, Laird, & Rubin, 1977; Jackson, 1991; Ng & Han, 1994; Berry, Dumais, & O'Brien, 1995; Cheeseman & Stutz, 1996; Ester et al., 1996; Guha, Rastogi, & Shim, 1998; Boley, 1998; Guha, Rastogi, & Shim, 1999; Karypis, Han, & Kumar, 1999a; Strehl & Ghosh, 2000; Ding et al., 2001). Two recent surveys on the topics (Jain, Murty, & Flynn, 1999; Han, Kamber, & Tung, 2001) offer a comprehensive summary of the different applications and algorithms. These algorithms can be categorized along different dimensions based either on the underlying methodology of the algorithm, leading to *agglomerative* or *partitional* approaches, or on the structure of the final solution, leading to *hierarchical* or *non-hierarchical* solutions.

In recent years, various researchers have recognized that partitional clustering algorithms are well-suited for clustering large document datasets due to their relatively low computational requirements (Cutting et al., 1992; Larsen & Aone, 1999; Steinbach, Karypis, &

*This work was supported by NSF ACI-0133464, CCR-9972519, EIA-9986042, ACI-9982274, and by Army HPC Research Center contract number DAAH04-95-C-0008.

Kumar, 2000). A key characteristic of many partitional clustering algorithms is that they use a global criterion function whose optimization drives the entire clustering process. For some of these algorithms the criterion function is implicit (e.g., PDDP (Boley, 1998)), whereas for other algorithms (e.g, K-means (MacQueen, 1967), Cobweb (Fisher, 1987), and Autoclass (Cheeseman & Stutz, 1996)) the criterion function is explicit and can be easily stated. This latter class of algorithms can be thought of as consisting of two key components. First is the criterion function that the clustering solution optimizes, and second is the actual algorithm that achieves this optimization.

The focus of this paper is to study the suitability of different criterion functions to the problem of clustering document datasets. In particular, we evaluate a total of seven criterion functions that measure various aspects of intra-cluster similarity, inter-cluster dissimilarity, and their combinations. These criterion functions utilize different views of the underlying collection by either modeling the documents as vectors in a high dimensional space or by modeling the collection as a graph. We experimentally evaluated the performance of these criterion functions using 15 different datasets obtained from various sources. Our experiments show that different criterion functions do lead to substantially different results and that there are a set of criterion functions that produce the best clustering solutions.

Our analysis of the different criterion functions shows that their overall performance depends on the degree to which they can correctly operate when the dataset contains clusters of different tightness (i.e., they contain documents whose average pairwise similarities are different) and the degree to which they can produce balanced clusters. Moreover, our analysis also shows that the sensitivity to the difference in the cluster tightness can also explain an outcome of our study (that was also observed in earlier results reported in Steinbach, Karypis, and Kumar (2000)), that for some clustering algorithms the solution obtained by performing a sequence of repeated bisections is better (and for some criterion functions by a considerable amount) than the solution obtained by computing the clustering directly. When the solution is computed via repeated bisections, the tightness difference between the two clusters that are discovered is in general smaller than the tightness differences between all the clusters. As a result, criterion functions that cannot handle well variation in cluster tightness tend to perform substantially better when used to compute the clustering via repeated bisections.

The rest this paper is organized as follows. Section 2 provides some information on the document representation and similarity measure used in our study. Section 3 describes the different criterion functions and the algorithms used to optimize them. Section 4 provides the detailed experimental evaluation of the various criterion functions. Section 5 analyzes the different criterion functions and explains their performance. Finally, Section 6 provides some concluding remarks.

2. Preliminaries

Document representation. The various clustering algorithms described in this paper represent each document using the well-known *term frequency-inverse document frequency* (tf-idf) vector-space model (Salton, 1989). In this model, each document d is considered to

be a vector in the term-space and is represented by the vector

$$d_{tfidf} = (tf_1 \log(n/df_1), tf_2 \log(n/df_2), \ldots, tf_m \log(n/df_m)),$$

where tf_i is the frequency of the ith term (i.e., term frequency), n is the total number of documents, and df_i is the number of documents that contain the ith term (i.e., document frequency). To account for documents of different lengths, the length of each document vector is normalized so that it is of unit length. In the rest of the paper, we will assume that the vector representation for each document has been weighted using *tf-idf* and normalized so that it is of unit length.

Similarity measures. Two prominent ways have been proposed to compute the similarity between two documents d_i and d_j. The first method is based on the commonly-used (Salton, 1989) cosine function

$$\cos(d_i, d_j) = d_i{}^t d_j / (\|d_i\| \|d_j\|),$$

and since the document vectors are of unit length, it simplifies to $d_i{}^t d_j$. The second method computes the similarity between the documents using the Euclidean distance $\text{dis}(d_i, d_j) = \|d_i - d_j\|$. Note that besides the fact that one measures similarity and the other measures distance, these measures are quite similar to each other because the document vectors are of unit length.

Definitions. Throughout this paper we will use the symbols n, m, and k to denote the number of documents, the number of terms, and the number of clusters, respectively. We will use the symbol S to denote the set of n documents to be clustered, $S_1, S_2, \ldots, S_k$ to denote each one of the k clusters, and $n_1, n_2, \ldots, n_k$ to denote their respective sizes. Given a set A of documents and their corresponding vector representations, we define the ***composite*** vector D_A to be $D_A = \sum_{d \in A} d$, and the ***centroid*** vector C_A to be $C_A = D_A/|A|$.

3. Document clustering

At a high-level the problem of clustering is defined as follows. Given a set S of n documents, we would like to partition them into a pre-determined number of k subsets $S_1, S_2, \ldots, S_k$, such that the documents assigned to each subset are more similar to each other than the documents assigned to different subsets.

As discussed in the introduction, our focus is to study the suitability of various clustering criterion functions in the context of partitional document clustering algorithms. Consequently, given a particular clustering criterion function $\mathcal{C}$, the clustering problem is to compute a k-way clustering solution such that the value of $\mathcal{C}$ is optimized. In the rest of this section we first present a number of different criterion functions that can be used to both evaluate and drive the clustering process, followed by a description of the algorithms that were used to perform their optimization.

Table 1. Clustering criterion functions.

$\mathcal{I}_1$	maximize	$\sum_{r=1}^{k} n_r \left(\frac{1}{n_r^2} \sum_{d_i, d_j \in S_r} \cos(d_i, d_j) \right) = \sum_{r=1}^{k} \frac{\|D_r\|^2}{n_r}$	(1)
$\mathcal{I}_2$	maximize	$\sum_{r=1}^{k} \sum_{d_i \in S_r} \cos(d_i, C_r) = \sum_{r=1}^{k} \|D_r\|$	(2)
$\mathcal{E}_1$	minimize	$\sum_{r=1}^{k} n_r \cos(C_r, C) = \sum_{r=1}^{k} n_r \frac{{D_r}^t D}{\|D_r\|}$	(3)
$\mathcal{H}_1$	maximize	$\frac{\mathcal{I}_1}{\mathcal{E}_1} = \frac{\sum_{r=1}^{k} \|D_r\|^2 / n_r}{\sum_{r=1}^{k} n_r {D_r}^t D / \|D_r\|}$	(4)
$\mathcal{H}_2$	maximize	$\frac{\mathcal{I}_2}{\mathcal{E}_1} = \frac{\sum_{r=1}^{k} \|D_r\|}{\sum_{r=1}^{k} n_r {D_r}^t D / \|D_r\|}$	(5)
$\mathcal{G}_1$	minimize	$\sum_{r=1}^{k} \frac{\text{cut}(S_r, S - S_r)}{\sum_{d_i, d_j \in S_r} \cos(d_i, d_j)} = \sum_{r=1}^{k} \frac{{D_r}^t (D - D_r)}{\|D_r\|^2}$	(6)
$\mathcal{G}_2$	minimize	$\sum_{r=1}^{k} \frac{\text{cut}(V_r, V - V_r)}{W(V_r)}$	(7)

3.1. Clustering criterion functions

Our study involves a total of seven different clustering criterion functions that are defined in Table 1. These functions optimize various aspects of intra-cluster similarity, inter-cluster dissimilarity, and their combinations, and represent some of the most widely-used criterion functions for document clustering.

The $\mathcal{I}_1$ criterion function (Eq. (1)) maximizes the sum of the average pairwise similarities (as measured by the cosine function) between the documents assigned to each cluster weighted according to the size of each cluster and has been used successfully for clustering document datasets (Puzicha, Hofmann, & Buhmann, 2000). The $\mathcal{I}_2$ criterion function (Eq. (2)) is used by the popular vector-space variant of the K-means algorithm (Cutting et al., 1992; Larsen & Aone, 1999; Dhillon & Modha, 2001; Steinbach, Karypis, & Kumar, 2000). In this algorithm each cluster is represented by its centroid vector and the goal is to find the solution that maximizes the similarity between each document and the centroid of the cluster that is assigned to. Comparing $\mathcal{I}_1$ and $\mathcal{I}_2$ we see that the essential difference between them is that $\mathcal{I}_2$ scales the within-cluster similarity by the $\|D_r\|$ term as opposed to the n_r term used by $\mathcal{I}_1$. $\|D_r\|$ is the square-root of the pairwise similarity between all the document in S_r and will tend to emphasize clusters whose documents have smaller pairwise similarities compared to clusters with higher pairwise similarities.

The $\mathcal{E}_1$criterion function (Eq. (3)) computes the clustering by finding a solution that separates the documents of each cluster from the entire collection. Specifically, it tries to minimize the cosine between the centroid vector of each cluster and the centroid vector

of the entire collection. The contribution of each cluster is weighted proportionally to its size so that larger clusters will be weighted higher in the overall clustering solution. $\mathcal{E}_1$ was motivated by multiple discriminant analysis and is similar to minimizing the trace of the between-cluster scatter matrix (Duda, Hart, & Stork, 2001).

The $\mathcal{H}_1$ and $\mathcal{H}_2$ criterion functions (Eqs. (4) and (5)) are obtained by combining criterion $\mathcal{I}_1$ with $\mathcal{E}_1$, and $\mathcal{I}_2$ with $\mathcal{E}_1$, respectively. Since $\mathcal{E}_1$ is minimized, both $\mathcal{H}_1$ and $\mathcal{H}_2$ need to be maximized as they are inversely related to $\mathcal{E}_1$.

The criterion functions that we described so far, view each document as a multidimensional vector. An alternate way of modeling the relations between documents is to use graphs. Two types of graphs are commonly-used in the context of clustering. The first corresponds to the document-to-document similarity graph G_s and the second to the document-to-term bipartite graph G_b (Beeferman & Berger, 2000; Zha et al., 2001a; Dhillon, 2001). G_s is obtained by treating the pairwise similarity matrix of the dataset as the adjacency matrix of G_s, whereas G_b is obtained by viewing the documents and the terms as the two sets of vertices (V_d and V_t) of a bipartite graph. In this bipartite graph, if the ith document contains the jth term, then there is an edge connecting the corresponding ith vertex of V_d to the jth vertex of V_t. The weights of these edges are set using the *tf-idf* model discussed in Section 2.

Viewing the documents in this fashion, a number of edge-cut-based criterion functions can be used to cluster document datasets (Cheng & Wei, 1991; Hagen & Kahng, 1991; Shi & Malik, 2000; Ding et al., 2001; Zha et al., 2001a; Dhillon, 2001). $\mathcal{G}_1$ and $\mathcal{G}_2$ (Eqs. (6) and (7)) are two such criterion functions that are defined on the similarity and bipartite graphs, respectively. The $\mathcal{G}_1$ function (Ding et al., 2001) views the clustering process as that of partitioning the documents into groups that minimize the edge-cut of each partition. However, because this edge-cut-based criterion function may have trivial solutions the edge-cut of each cluster is scaled by the sum of the cluster's internal edges (Ding et al., 2001). Note that cut $(S_r, S - S_r)$ in Eq. (6) is the edge-cut between the vertices in S_r and the rest of the vertices $S - S_r$, and can be re-written as ${D_r}^t(D - D_r)$ since the similarity between documents is measured using the cosine function. The $\mathcal{G}_2$ criterion function (Zha et al., 2001a; Dhillon, 2001) views the clustering problem as a simultaneous partitioning of the documents and the terms so that it minimizes the normalized edge-cut (Shi & Malik, 2000) of the partitioning. Note that V_r is the set of vertices assigned to the rth cluster and $W(V_r)$ is the sum of the weights of the adjacency lists of the vertices assigned to the rth cluster.

3.2. *Criterion function optimization*

There are many techniques that can be used to optimize the criterion functions described in the previous section. They include relatively simple greedy schemes, iterative schemes with varying degree of hill-climbing capabilities, and powerful but computationally expensive spectral-based optimizers (MacQueen, 1967; Cheeseman & Stutz, 1996; Fisher, 1996; Meila & Heckerman, 2001; Karypis, Han, & Kumar, 1999b; Boley, 1998; Zha et al., 2001b; Zha et al., 2001a; Dhillon, 2001). Despite this wide-range of choices, in our study, the various criterion functions were optimized using a simple and obvious greedy strategy. This was primarily motivated by our experience with document datasets (and similar results presented in Savaresi and Boley (2001)), which showed that greedy-based schemes (when

run multiple times) produce comparable results to those produced by more sophisticated optimization algorithms for the range of the number of clusters that we used in our experiments. Nevertheless, the choice of the optimization methodology can potentially impact the relative performance of the various criterion functions, since that performance may depend on the optimizer (Fisher, 1996). However, as we will see later in Section 5, our analysis of the criterion functions correlates well with our experimental results, suggesting that the choice of the optimizer does not appear to be biasing the experimental comparisons.

Our greedy optimizer computes the clustering solution by first obtaining an initial k-way clustering and then applying an iterative refinement algorithm to further improve it. During initial clustering, k documents are randomly selected to form the *seeds* of the clusters and each document is assigned to the cluster corresponding to its most similar seed. This approach leads to an initial clustering solution for all but the $\mathcal{G}_2$ criterion function as it does not produce an initial partitioning for the vertices corresponding to the terms (V_t). The initial partitioning of V_t is obtained by assigning each term v to the partition that is most connected with. The iterative refinement strategy that we used is based on the *incremental* refinement scheme described in Duda, Hart, and Stork (2001). During each iteration, the documents are visited in a random order and each document is moved to the cluster that leads to the highest improvement in the value of the criterion function. If no such cluster exists, then the document does not move. The refinement phase ends, as soon as an iteration is performed in which no documents were moved between clusters. Note that in the case of $\mathcal{G}_2$, the refinement algorithm alternates between document-vertices and term-vertices (Kolda & Hendrickson, 2000).

The algorithms used during the refinement phase are greedy in nature, they are not guaranteed to converge to a global optimum, and the local optimum solution they obtain depends on the particular set of seed documents that were selected to obtain the initial clustering. To eliminate some of this sensitivity, the overall process is repeated a number of times. That is, we compute N different clustering solutions (i.e., initial clustering followed by cluster refinement), and the one that achieves the best value for the particular criterion function is kept. In all of our experiments, we used $N = 10$. For the rest of this discussion when we refer to a clustering solution we will mean the solution that was obtained by selecting the best (with respect to the value of the respective criterion function) out of these N potentially different solutions.

4. Experimental results

We experimentally evaluated the performance of the different clustering criterion functions on a number of different datasets. In the rest of this section we first describe the various datasets and our experimental methodology, followed by a description of the experimental results.

4.1. Document collections

In our experiments, we used a total of 15 datasets (http://www.cs.umn.edu/~karypis/cluto/files /datasets.tar.gz.) whose general characteristics and sources are summarized in Table 2.

Table 2. Summary of datasets used to evaluate the various clustering criterion functions.

Data	Source	No. of documents	No. of terms	No. of classes
classic	CACM/CISI/CRANFIELD/MEDLINE (ftp://ftp.cs.cornell.edu/pub/smart)	7089	12009	4
fbis	FBIS (TREC-5 (TREC, 1999))	2463	12674	17
hitech	San Jose Mercury (TREC, TIPSTER Vol. 3)	2301	13170	6
reviews	San Jose Mercury (TREC, TIPSTER Vol. 3)	4069	23220	5
sports	San Jose Mercury (TREC, TIPSTER Vol. 3)	8580	18324	7
la12	LA Times (TREC-5 (TREC, 1999))	6279	21604	6
new3	TREC-5 & TREC-6 (TREC, 1999)	9558	36306	44
tr31	TREC-5 & TREC-6 (TREC, 1999)	927	10128	7
tr41	TREC-5 & TREC-6 (TREC, 1999)	878	7454	10
ohscal	OHSUMED-233445 (Hersh et al., 1994)	11162	11465	10
re0	Reuters-21578 (Lewis, 1999)	1504	2886	13
re1	Reuters-21578 (Lewis, 1999)	1657	3758	25
k1a	WebACE (Han et al., 1998)	2340	13879	20
k1b	WebACE (Han et al., 1998)	2340	13879	6
wap	WebACE (Han et al., 1998)	1560	8460	20

The smallest of these datasets contained 878 documents and the largest contained 11,162 documents. To ensure diversity in the datasets, we obtained them from different sources. For all datasets we used a stop-list to remove common words and the words were stemmed using Porter's suffix-stripping algorithm (Porter, 1980). Moreover, any term that occurs in fewer than two documents was eliminated.

4.2. Experimental methodology and metrics

For each one of the different datasets we obtained a 5-, 10-, 15-, and 20-way clustering solution that optimized the various clustering criterion functions shown in Table 1. The quality of a clustering solution was evaluated using the ***entropy*** measure that is based on how the various classes of documents are distributed within each cluster. Given a particular cluster S_r of size n_r, the entropy of this cluster is defined to be

$$E(S_r) = -\frac{1}{\log q} \sum_{i=1}^{q} \frac{n_r^i}{n_r} \log \frac{n_r^i}{n_r},$$

where q is the number of classes in the dataset and n_r^i is the number of documents of the ith class that were assigned to the rth cluster. The entropy of the entire solution is defined

to be the sum of the individual cluster entropies weighted according to the cluster size, i.e.,

$$Entropy = \sum_{r=1}^{k} \frac{n_r}{n} E(S_r).$$

A perfect clustering solution will be the one that leads to clusters that contain documents from only a single class, in which case the entropy will be zero. In general, the smaller the entropy values, the better the clustering solution is.

To eliminate any instances that a particular clustering solution for a particular criterion function got trapped into a bad local optimum, in all of our experiments we found ten different clustering solutions. As discussed in Section 3.2 each of these ten clustering solutions correspond to the best solution (in terms of the respective criterion function) out of ten different initial partitioning and refinement phases. As a result, for each particular value of k and criterion function we generated 100 different clustering solutions. The overall number of experiments that we performed was $3 * 100 * 4 * 8 * 15 = 144{,}000$, that were completed in about 8 days on a Pentium III@600MHz workstation.

One of the problems associated with such large-scale experimental evaluation is that of summarizing the results in a meaningful and unbiased fashion. Our summarization is done as follows. For each dataset and value of k, we divided the entropy obtained by a particular criterion function by the smallest entropy obtained for that particular dataset and value of k over the different criterion functions. These ratios represent the degree to which a particular criterion function performed worse than the best criterion function for that dataset and value of k. These ratios are less sensitive to the actual entropy values and the particular value of k. We will refer to these ratios as ***relative entropies***. Now, for each criterion function and value of k we averaged these relative entropies over the various datasets. A criterion function that has an ***average relative entropy*** close to 1.0 indicates that this function did the best for most of the datasets. On the other hand, if the average relative entropy is high, then this criterion function performed poorly. In addition to these numerical averages, we evaluated the statistical significance of the relative performance of the criterion functions using a paired-t test (Devore & Peck, 1997) based on the original entropies for each dataset. The original entropy values for all the experiments presented in this paper can be found in Zhao and Karypis (2001).

4.3. Evaluation of direct k-way clustering

Our first set of experiments was focused on evaluating the quality of the clustering solutions produced by the various criterion functions when they were used to compute a k-way clustering solution directly. The values for the average relative entropies for the 5-, 10-, 15-, and 20-way clustering solutions are shown in Table 3. The row labeled "Avg" contains the average of these averages over the four sets of solutions. Furthermore, the last column shows the relative ordering of the different schemes using the paired-t test.

From these results we can see that the $\mathcal{I}_1$ and the $\mathcal{G}_2$ criterion functions lead to clustering solutions that are consistently worse (in the range of 19–35%) than the solutions obtained

Table 3. Average relative entropies for the clustering solutions obtained via direct k-way clustering and their statistical significance.

k	$\mathcal{I}_1$	$\mathcal{I}_2$	$\mathcal{E}_1$	$\mathcal{H}_1$	$\mathcal{H}_2$	$\mathcal{G}_1$	$\mathcal{G}_2$	Statistical significance test, p-value = .05
5	1.361	1.041	1.044	1.069	**1.033**	1.092	1.333	$(\mathcal{I}_1, \mathcal{G}_2) \ll (\mathcal{G}_1, \mathcal{H}_1) \ll (\mathcal{E}_1, \mathcal{I}_2, \mathcal{H}_2)$
10	1.312	1.042	1.069	**1.035**	1.040	1.148	1.380	$\mathcal{G}_2 \ll \mathcal{I}_1 \ll \mathcal{G}_1 \ll (\mathcal{E}_1, \mathcal{H}_2, \mathcal{H}_1, \mathcal{I}_2)$
15	1.252	**1.019**	1.071	1.029	1.029	1.132	1.402	$\mathcal{G}_2 \ll \mathcal{I}_1 \ll \mathcal{G}_1 \ll \mathcal{E}_1 \ll (\mathcal{H}_2, \mathcal{H}_1, \mathcal{I}_2)$
20	1.236	**1.018**	1.086	1.022	1.035	1.139	1.486	$\mathcal{G}_2 \ll \mathcal{I}_1 \ll \mathcal{G}_1 \ll \mathcal{E}_1 \ll \mathcal{H}_2 \ll (\mathcal{I}_2, \mathcal{H}_1)$
Avg	1.290	**1.030**	1.068	1.039	1.034	1.128	1.400	

Underlined entries represent the best performing scheme in terms of average relative entropies. Note that "$\ll$" indicates that schemes on the right are significantly better than the schemes on the left, and "( )" indicates that the relationship is not significant. The order of the schemes within parentheses represent the order of the weak relationship.

using the other criterion functions. On the other hand, the $\mathcal{I}_2$, $\mathcal{H}_2$, and $\mathcal{H}_1$ criterion functions lead to the best solutions irrespective of the number of clusters. Over the entire set of experiments, these methods are either the best or always within 2% of the best solution. Finally, $\mathcal{E}_1$ performs the next best followed by $\mathcal{G}_1$ that produces solutions whose average relative entropy is 9% worse than those produced by the best scheme.

4.4. Evaluation of k-way clustering via repeated bisections

Our second set of experiments was focused on evaluating the clustering solutions produced by the various criterion functions when the overall solution was obtained via a sequence of cluster bisections (RB). In this approach, a k-way solution is obtained by first bisecting the entire collection. Then, one of the two clusters is selected and it is further bisected, leading to a total of three clusters. This step of selecting and bisecting a cluster is performed $k - 1$ times leading to the desired k-way clustering solution. Each of these bisections is performed so that the resulting bisection optimizes a particular criterion function. However, the overall k-way clustering solution will not necessarily be at a local optimum with respect to that criterion function.

The key step in this algorithm is the method used to select which cluster to bisect next and a number of different approaches were described in Steinbach, Karypis, and Kumar (2000), Karypis and Han (2000) and Savaresi et al. (2002). In all of our experiments, we selected the largest cluster, as this approach leads to reasonably good and balanced clustering solutions (Steinbach, Karypis, & Kumar, 2000).

The average relative entropies of the resulting solutions are shown in Table 4, and these results are in general consistent with those obtained for direct k-way clustering (Table 3). The $\mathcal{I}_1$ and $\mathcal{G}_2$ functions lead to the worst solutions, $\mathcal{H}_2$ leads to the best overall solutions, and $\mathcal{I}_2$, $\mathcal{E}_1$, and $\mathcal{G}_1$ are within 2% of the best. However, in the case of RB, there is a reduction in the relative difference between the best and the worst schemes. For example, $\mathcal{G}_2$ is only 13% worse than the best (compared to 35% for direct k-way). Similar trends can be observed for the other functions. This relative improvement becomes most apparent

Table 4. Average relative entropies for the clustering solutions obtained via repeated bisections and their statistical significance.

k	$\mathcal{I}_1$	$\mathcal{I}_2$	$\mathcal{E}_1$	$\mathcal{H}_1$	$\mathcal{H}_2$	$\mathcal{G}_1$	$\mathcal{G}_2$	Statistical significance test, p-value = .05
5	1.207	1.050	1.060	1.083	**1.049**	1.053	1.191	$(\mathcal{I}_1, \mathcal{G}_2) \ll (\mathcal{H}_1, \mathcal{E}_1, \mathcal{G}_1, \mathcal{I}_2, \mathcal{H}_2)$
10	1.243	1.112	1.083	1.129	**1.056**	1.106	1.221	$(\mathcal{I}_1, \mathcal{G}_2) \ll (\mathcal{H}_1, \mathcal{I}_2, \mathcal{G}_1, \mathcal{E}_1, \mathcal{H}_2)$
15	1.190	1.085	1.077	1.102	**1.079**	1.085	1.205	$(\mathcal{G}_2, \mathcal{I}_1) \ll (\mathcal{H}_1, \mathcal{G}_1, \mathcal{E}_1, \mathcal{I}_2, \mathcal{H}_2)$
20	1.183	1.070	**1.057**	1.085	1.072	1.075	1.209	$(\mathcal{G}_2, \mathcal{I}_1) \ll (\mathcal{H}_1, \mathcal{G}_1, \mathcal{E}_1, \mathcal{I}_2, \mathcal{H}_2)$
Avg	1.206	1.079	1.069	1.100	**1.064**	1.080	1.207	

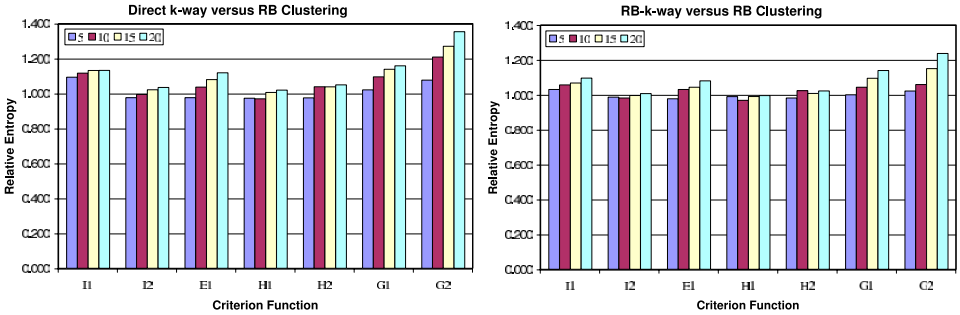

Figure 1. (*left chart*) The relative performance of direct k-way clustering over that of repeated bisections (RB). (*right chart*) The relative performance of repeated bisections-based clustering followed by k-way refinement over that of repeated bisections alone. These results correspond to averages over the different datasets.

for the $\mathcal{G}_1$ criterion function that now almost always performs within 2% of the best. The reason for these improvements will be discussed in Section 5.

Figure 1 compares the quality of the solutions obtained via direct k-way to those obtained via repeated bisections. These plots were obtained by dividing the entropies of the solutions obtained by the direct k-way approach with those obtained by the RB approach and averaging them over the fifteen datasets. Ratios that are greater than one indicate that the RB approach leads to better solutions than direct k-way and vice versa. From these plots we see that the direct k-way solutions obtained by $\mathcal{I}_1$, $\mathcal{G}_1$, and $\mathcal{G}_2$ are worse than those obtained by RB clustering. For the remaining functions, the relative performance appears to be sensitive to the number of clusters. For small number of clusters, the direct approach tends to lead to better solutions; however, as the number of clusters increases the RB approach tends to outperform it. In fact, the sensitivity on k appears to be true for all seven criterion functions, and the main difference has to do with how quickly the relative quality of the direct k-way clustering solution degrades. Among the different functions, $\mathcal{I}_2$, $\mathcal{H}_1$, and $\mathcal{H}_2$ appear to be the least sensitive as their relative performance does not change significantly between the two clustering methods as k increases.

Table 5. Average relative entropies for the clustering solutions obtained via repeated bisections followed by k-way refinement and their statistical significance.

k	$\mathcal{I}_1$	$\mathcal{I}_2$	$\mathcal{E}_1$	$\mathcal{H}_1$	$\mathcal{H}_2$	$\mathcal{G}_1$	$\mathcal{G}_2$	Statistical significance test, p-value = .05
5	1.304	1.081	1.077	1.121	**1.076**	1.097	1.273	$(\mathcal{I}_1, \mathcal{G}_2) \ll (\mathcal{H}_1, \mathcal{G}_1, \mathcal{E}_1, \mathcal{I}_2, \mathcal{H}_2)$
10	1.278	1.065	1.088	1.063	**1.051**	1.127	1.255	$(\mathcal{G}_2, \mathcal{I}_1) \ll \mathcal{G}_1 \ll (\mathcal{E}_1, \mathcal{H}_1, \mathcal{I}_2, \mathcal{H}_2)$
15	1.234	**1.037**	1.089	1.057	1.046	1.140	1.334	$\mathcal{G}_2 \ll \mathcal{I}_1 \ll (\mathcal{G}_1, \mathcal{E}_1) \ll (\mathcal{H}_1, \mathcal{H}_2, \mathcal{I}_2)$
20	1.248	**1.030**	1.098	1.041	1.051	1.164	1.426	$\mathcal{G}_2 \ll \mathcal{I}_1 \ll \mathcal{G}_1 \ll \mathcal{E}_1 \ll (\mathcal{H}_2, \mathcal{H}_1, \mathcal{I}_2)$
Avg	1.266	**1.053**	1.088	1.070	1.056	1.132	1.322	

4.5. Evaluation of k-way clustering via repeated bisections followed by k-way refinement

To further investigate the behavior of the RB-based clustering approach we performed a sequence of experiments in which the final solution obtained by the RB approach for a particular criterion function was further refined using the greedy k-way refinement algorithm described in Section 3.2. We will refer to this scheme as *RB-k-way*. The average relative entropies for this set of experiments are shown in Table 5.

Comparing the relative performance of the various criterion functions we can see that they are more similar to those of direct k-way (Table 3) than those of the RB-based approach (Table 4). In particular, $\mathcal{I}_2$, $\mathcal{E}_1$, $\mathcal{H}_1$, and $\mathcal{H}_2$ tend to outperform the rest, with $\mathcal{I}_2$ performing the best. Also, we can see that both $\mathcal{I}_1$, $\mathcal{G}_1$, and $\mathcal{G}_2$ are considerably worse than the best scheme. Figure 1 compares the relative quality of the RB-k-way solutions to the solutions obtained by the RB-based scheme. Looking at these results we can see that by optimizing the $\mathcal{I}_1$, $\mathcal{E}_1$, $\mathcal{G}_1$, and $\mathcal{G}_2$ criterion functions, the quality of the solutions become worse, especially for large number of clusters. The largest degradation happens for $\mathcal{G}_1$and $\mathcal{G}_2$. On the other hand, as we optimize either $\mathcal{I}_2$, $\mathcal{H}_1$, or $\mathcal{H}_2$, the overall cluster quality changes only slightly (sometimes it gets better and sometimes it gets worse). These results verify the observations we made in Section 4.4 that suggest that the optimization of some of the criterion functions does not necessarily lead to better quality clusters, especially for large values of k.

5. Discussion and analysis

The experiments presented in Section 4 showed two interesting trends. First, the quality of the solutions produced by some seemingly similar criterion functions is often substantially different. For instance, both $\mathcal{I}_1$ and $\mathcal{I}_2$ find clusters by maximizing a particular within cluster similarity function. However, $\mathcal{I}_2$ performs substantially better than $\mathcal{I}_1$. This is also true for $\mathcal{E}_1$ and $\mathcal{G}_1$ that attempt to minimize a function that takes into account both the within cluster similarity and the across cluster dissimilarity. However, in most of the experiments, $\mathcal{E}_1$ tends to perform consistently better than $\mathcal{G}_1$. The second trend is that for many criterion functions, the quality of the solutions produced via repeated bisections is better than the corresponding solutions produced either via direct k-way clustering or after performing k-way refinement. Furthermore, this performance gap seems to increase with the number of clusters k. In the

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	1035	0.098	1034	1					
2	594	0.125		1	592				1
3	322	0.191		321	1				
4	653	0.127		1		652			
5	413	0.163	413						
6	1041	0.058			1041				
7	465	0.166	464		1				
8	296	0.172		296					
9	3634	0.020	1393	789	694	157	121	145	335
10	127	0.268	108	1	17		1		

$\mathcal{I}_1$ Criterion Function (Entropy=0.357)

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	475	0.087	97	35	143	8	112	64	16
2	384	0.129	1	1		381			1
3	1508	0.032	310	58	1055	11	5	59	10
4	844	0.094	1	1	841				1
5	400	0.163		1		399			
6	835	0.097	829		6				
7	1492	0.067	1489	1	2				
8	756	0.099	2	752	1	1			
9	621	0.108	618	1	2				
10	1265	0.036	65	560	296	9	5	22	308

$\mathcal{I}_2$ Criterion Function (Entropy=0.240)

Figure 2. The cluster-class distribution of the clustering solutions for the $\mathcal{I}_1$ and $\mathcal{I}_2$ criterion functions for the *sports* dataset.

remainder of this section we present an analysis that explains the cause of these trends. Our analyses are specific to selected criterion functions, and thus may have limited direct transfer in cases where other criteria are used. However, we believe that such analyses of criteria biases are important generally to better understand empirical findings. This is particularly important in clustering studies, an area in which a plethora of criteria exist, some appearing quite similar in form, but with very different implications for clustering results.

5.1. *Analysis of the $\mathcal{I}_1$ and $\mathcal{I}_2$ criterion functions*

As a starting point for analyzing the $\mathcal{I}_1$ and $\mathcal{I}_2$ criterion functions it is important to qualitatively understand the solutions that they produce. Figure 2 shows the 10-way clustering solutions obtained for the *sports* dataset using the direct clustering approach for $\mathcal{I}_1$ and $\mathcal{I}_2$. The rows of each subtable represent a particular cluster, and show the class distribution of the documents assigned to it. The columns labeled "Size" show the number of documents assigned to each cluster and those labeled "Sim" show the average pairwise similarity between the documents of each cluster. From these results we can see that both $\mathcal{I}_1$ and $\mathcal{I}_2$ produce solutions that contain a mixture of large, loose clusters and small, tight clusters. However, $\mathcal{I}_1$ behaves differently from $\mathcal{I}_2$ in two ways. (i) $\mathcal{I}_1$'s solution has a cluster (*cid* = *9*) that contains a very large number of documents from different categories and very low average pairwise similarities, whereas $\mathcal{I}_2$'s solution does not. This is also the reason why $\mathcal{I}_1$'s solution has a higher overall entropy value compared to $\mathcal{I}_2$'s (0.357 *vs* 0.240). (ii) Excluding this large poor cluster, $\mathcal{I}_1$'s remaining clusters tend to be quite pure and relatively *tight* (i.e., high "Sim" values), whereas $\mathcal{I}_2$'s clusters are somewhat less pure and less tight. The above observations on the characteristics of the solutions produced by $\mathcal{I}_1$ and $\mathcal{I}_2$ and the reasons as to why the former leads to higher entropy solutions hold for the remaining datasets as well.

To analyze this behavior we focus on the properties of an optimal clustering solution with respect to either $\mathcal{I}_1$ or $\mathcal{I}_2$ and show how the tightness of each cluster affects the assignment

of documents between the clusters. The following two propositions, whose proofs are in Appendix A, state the properties that are satisfied by the optimal solutions produced by the $\mathcal{I}_1$ and $\mathcal{I}_2$ criterion functions:

Proposition 1. *Given an optimal k-way solution* $\{S_1, S_2, \ldots, S_k\}$ *with respect to* $\mathcal{I}_1$, *then for each pair of clusters* S_i *and* S_j, *each document* $d \in S_i$ *satisfies the following inequality*:

$$\delta_i - \delta_j \geq \frac{\mu_i - \mu_j}{2}, \tag{8}$$

where μ_i *is the average pairwise similarity between the document of* S_i *excluding* d, δ_i *is the average pairwise similarity between* d *and the other documents of* S_i, μ_j *is the average pairwise similarity between the document of* S_j, *and* δ_j *is the average pairwise similarity between* d *and the documents of* S_j.

Proposition 2. *Given an optimal k-way solution* $\{S_1, S_2, \ldots, S_k\}$ *with respect to* $\mathcal{I}_2$, *then for each pair of clusters* S_i *and* S_j, *each document* $d \in S_i$ *satisfies the following inequality*:

$$\frac{\delta_i}{\delta_j} \geq \sqrt{\frac{\mu_i}{\mu_j}}, \tag{9}$$

where μ_i, μ_j, δ_i, *and* δ_j *is as defined in Proposition* 1.

From Eq. (8) with (9), we have that if the optimal solution contains clusters with substantially different tightness, then both criterion functions lead to optimal solutions in which documents that are more similar to a tighter cluster are assigned to a looser cluster. That is, without loss of generality, if $\mu_i > \mu_j$, then a document for which δ_i is small will be assigned to S_j, even if $\delta_j < \delta_i$. However, what differentiates the two criterion functions is how small δ_j can be relative to δ_i before such an assignment can take place. In the case of $\mathcal{I}_1$, even if $\delta_j = 0$ (i.e., document d has *nothing* in common with the documents of S_j), d can still be assigned to S_j as long as $\delta_i < (\mu_i - \mu_j)/2$, i.e., d has a relatively low average similarity with the documents of S_i. On the other hand, $\mathcal{I}_2$ will only assign d to S_j if it has a non-trivial average similarity to the documents of S_j ($\delta_j > \delta_i\sqrt{\mu_j/\mu_i}$). In addition, when δ_i and δ_j are relatively small, that is

$$\delta_j < \mu_j \frac{\alpha - 1}{2(\sqrt{\alpha} - 1)} \quad \text{and} \quad \delta_i < \mu_i \frac{\sqrt{\alpha}(\alpha - 1)}{2(\sqrt{\alpha} - 1)}, \quad \text{where} \quad \alpha = \frac{\mu_i}{\mu_j},$$

for the same value of δ_j, $\mathcal{I}_1$ assigns documents to S_j that have higher δ_i values than $\mathcal{I}_2$ does. Of course whether or not such document assignments will happen, depends on the characteristics of the particular dataset, but as long as the dataset has such characteristics, regardless of how $\mathcal{I}_1$ or $\mathcal{I}_2$ are optimized, they will tend to converge to this type of solution.

These observations explain the results shown in figure 2, in which $\mathcal{I}_1$'s clustering solution contains nine fairly pure and tight clusters, and a single large and poor-quality cluster. That single cluster acts almost like a *garbage collector* that attracts all the peripheral documents of the other clusters.

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	1330	0.076	1327	2	1				
2	975	0.080	3	5	966				1
3	742	0.072	15	703	24				
4	922	0.079	84	8	32	797			1
5	768	0.078	760	1	6		1		
6	897	0.054	6	2	889				
7	861	0.091	845	0	15				1
8	565	0.079	24	525	13	1			2
9	878	0.034	93	128	114	4	97	121	321
10	642	0.068	255	36	286	7	24	24	10

$\mathcal{E}_1$ Criterion Function (Entropy=0.203)

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	519	0.146	516		3				
2	597	0.118	1		595				1
3	1436	0.033	53	580	357	13	100	20	313
4	720	0.105		718	1	1			
5	1664	0.032	1387	73	77	49	7	63	8
6	871	0.101	871						
7	1178	0.049	6	5	1167				
8	728	0.111		1		727			
9	499	0.133	498		1				
10	368	0.122	80	33	145	19	15	62	14

$\mathcal{G}_1$ Criterion (Entropy=0.239)

Figure 3. The cluster-class distribution of the clustering solutions for the $\mathcal{E}_1$ and $\mathcal{G}_1$ criterion functions for the *sports* dataset.

5.2. *Analysis of the $\mathcal{E}_1$ and $\mathcal{G}_1$ criterion functions*

Both $\mathcal{E}_1$ and $\mathcal{G}_1$ functions measure the quality of the overall clustering solution by taking into account the separation between clusters and the tightness of each cluster. However, as the experiments presented in Section 4 show $\mathcal{E}_1$ consistently leads to better solutions than $\mathcal{G}_1$. Figure 3 shows the 10-way clustering solutions produced by $\mathcal{E}_1$ and $\mathcal{G}_1$ for the *sports* dataset and illustrates this difference in the overall clustering quality. As we can see $\mathcal{E}_1$ finds clusters that are considerably more balanced than those produced by $\mathcal{G}_1$. In fact, the solution obtained by $\mathcal{G}_1$ exhibits similar characteristics (but to a lesser extent) with the corresponding solution obtained by the $\mathcal{I}_1$ criterion function described in the previous section. $\mathcal{G}_1$ tends to produce a mixture of large and small clusters, with the smaller clusters being reasonably tight and the larger clusters being quite loose.

In order to compare the $\mathcal{E}_1$ and $\mathcal{G}_1$ criterion functions it is important to rewrite them in a way that makes their similarities and dissimilarities apparent. To this end, let μ_r be the average similarity between the documents of the rth cluster S_r, and let ξ_r be the average similarity between the documents in S_r to the entire set of documents S. Using these definitions, the $\mathcal{E}_1$ and $\mathcal{G}_1$ functions (Eqs. (3) and (6)) can be rewritten as

$$\mathcal{E}_1 = \sum_{r=1}^{k} n_r \frac{{D_r}^t D}{\|D_r\|} = \sum_{r=1}^{k} n_r \frac{n_r n \xi_r}{n_r \sqrt{\mu_r}} = n \sum_{r=1}^{k} n_r \frac{\xi_r}{\sqrt{\mu_r}}, \tag{10}$$

$$\mathcal{G}_1 = \sum_{r=1}^{k} \frac{{D_r}^t (D - D_r)}{\|D_r\|^2} = \left(\sum_{r=1}^{k} \frac{n_r n \xi_r}{n_r^2 \mu_r} \right) - k = \left(n \sum_{r=1}^{k} \frac{1}{n_r} \frac{\xi_r}{\mu_r} \right) - k. \tag{11}$$

Note that since k in Eq. (11) is constant, it does not affect the overall solution and we will ignore it.

Comparing Eqs. (10) and (11) we can see that they differ on the way they measure the quality of a particular cluster, and on how they combine these individual cluster quality

measures to derive the overall quality of the clustering solution. In the case of $\mathcal{E}_1$, the quality of the rth cluster is measured as $\xi_r/\sqrt{\mu_r}$, whereas in the case of $\mathcal{G}_1$ it is measured as ξ_r/μ_r. Since the quality of each cluster is inversely related to either μ_r or $\sqrt{\mu_r}$, both measures will prefer solutions in which there are no clusters that are extremely loose. Because large clusters tend to have small μ_r values, both of the cluster quality measures will tend to produce solutions that contain reasonably balanced clusters. Furthermore, the sensitivity of $\mathcal{G}_1$'s cluster quality measure on clusters with small μ_r values is higher than the corresponding sensitivity of $\mathcal{E}_1$ ($\mu_r \leq \sqrt{\mu_r}$ because $\mu_r \leq 1$). Consequently, we would have expected $\mathcal{G}_1$ to lead to more balanced solutions than $\mathcal{E}_1$, which as the results in figure 3 show does not happen, suggesting that the second difference between $\mathcal{E}_1$ and $\mathcal{G}_1$ is the reason for the unbalanced clusters.

The $\mathcal{E}_1$ criterion function sums the individual cluster qualities weighting them proportionally to the size of each cluster. $\mathcal{G}_1$ performs a similar summation but each cluster quality is weighted proportionally to the *inverse* of the size of the cluster. This weighting scheme is similar to that used in the *ratio-cut* objective for graph partitioning (Cheng & Wei, 1991; Hagen & Kahng, 1991). Recall from our previous discussion that since the quality measure of each cluster is inversely related to μ_r, the quality measure of large clusters will have large values, as these clusters will tend to be loose (i.e., μ_r will be small). Now, in the case of $\mathcal{E}_1$, by multiplying the quality measure of a cluster by its size, it ensures that these large loose clusters contribute a lot to the overall value of $\mathcal{E}_1$'s criterion function. As a result, $\mathcal{E}_1$ will tend to be optimized when there are no large loose clusters. On the other hand, in the case of $\mathcal{G}_1$, by dividing the quality measure of a large loose cluster by its size, it has the net effect of decreasing the contribution of this cluster to the overall value of $\mathcal{G}_1$'s criterion function. As a result, $\mathcal{G}_1$ can be optimized at a point in which there exist some large and loose clusters.

5.3. *Analysis of the $\mathcal{G}_2$ criterion function*

The various experiments presented in Section 4 showed that the $\mathcal{G}_2$ criterion function consistently led to clustering solutions that were among the worst over the solutions produced by the other criterion functions. To illustrate how $\mathcal{G}_2$ fails, figure 4 shows the 10-way clustering solution that it produced via direct k-way clustering on the *sports* dataset. As we can see, $\mathcal{G}_2$ produces solutions that are highly unbalanced. For example, the sixth cluster contains over 2500 documents from many different categories, whereas the third cluster contains only 42 documents that are primarily from a single category. Note that, the clustering solution produced by $\mathcal{G}_2$ is very similar to that produced by the $\mathcal{I}_1$ criterion function (figure 2). In fact, for most of the clusters we can find a good one-to-one mapping between the two schemes.

The nature of $\mathcal{G}_2$'s criterion function makes it extremely hard to analyze it. However, one reason that can potentially explain the unbalanced clusters produced by $\mathcal{G}_2$ is the fact that it uses a normalized-cut inspired approach to combine the separation between the clusters (as measured by the cut) versus the size of the respective clusters. It has been shown in Ding et al. (2001) that when the normalized-cut approach is used in the context of traditional graph partitioning, it leads to a solution that is considerably more unbalanced than that

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	491	0.096	1	5	485				
2	1267	0.056	8	5	1244	10			
3	42	0.293	2	1	3		1	35	
4	630	0.113	0	627	2	1			
5	463	0.126	462		1				
6	2596	0.027	1407	283	486	184	42	107	87
7	998	0.040	49	486	124	8	79	3	249
8	602	0.120		1		601			
9	1202	0.081	1194	2	1	5			
10	289	0.198	289						

$\mathcal{G}_2$ Criterion Function (Entropy=0.315)

Figure 4. The cluster-class distribution of the clustering solutions for the $\mathcal{G}_2$ criterion function for the *sports* dataset.

obtained by the $\mathcal{G}_1$ criterion function. However, as our discussion in Section 5.2 showed, even $\mathcal{G}_1$'s balancing mechanism often leads to quite unbalanced clustering solutions.

5.4. *Analysis of direct k-way clustering versus repeated bisections*

From our analysis of the $\mathcal{I}_1$, $\mathcal{I}_2$, and $\mathcal{G}_1$ criterion functions we know that based on the difference between the tightness (i.e., the average pairwise similarity between the documents in the cluster) of the two clusters, documents that are naturally part of the tighter cluster will end up be assigned to the looser cluster. In other words, the various criterion functions will tend to produce incorrect clustering results when clusters have different degrees of tightness. Of course, the degree to which a particular criterion function is sensitive to tightness differences will be different for the various criterion functions. When the clustering solution is obtained via repeated bisections, the difference in tightness between each pair of clusters in successive bisections will tend to be relatively small. This is because, each cluster to be bisected, will tend to be relatively homogeneous (due to the way it was discovered), resulting in a pair of subclusters with small tightness differences. On the other hand, when the clustering is computed directly or when the final k-way clustering obtained via a sequence of repeated bisections is refined, there can exist clusters that have significant differences in tightness. Whenever such pairs of clusters occur, most of the criterion functions will end up moving some of the document of the tighter cluster (that are weakly connected to the rest of the documents in that cluster) to the looser cluster. Consequently, the final clustering solution can potentially be worse than that obtained via repeated bisections.

To illustrate this behavior we used the $\mathcal{I}_2$ criterion function and computed a 15-way clustering solution using repeated bisections and then refined it by performing a 15-way refinement for the *sports* dataset. These results are shown in figure 5. The RB solution contains some clusters that are quite loose and some clusters that are quite tight. Comparing

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	245	0.121	243	0	2				
2	596	0.067	2	1	593				
3	485	0.097	1	480	3	1			
4	333	0.080	3	6	3		2	1	318
5	643	0.104	642		1				
6	674	0.047	669	2	1	1	1		
7	762	0.099		1	760				1
8	826	0.045	42	525	247	6			6
9	833	0.105	832	1					
10	795	0.102	1	1	1	791			1
11	579	0.061	6		573				
12	647	0.034	174	34	156	10	119	144	10
13	191	0.110	189		2				
14	611	0.125	608		3				
15	360	0.168		359	1				

$\mathcal{I}_2$— RB (Entropy=0.125)

cid	Size	Sim	baseball	basketball	football	hockey	boxing	bicycling	golfing
1	292	0.120	280		11		1		
2	471	0.080	1	2	468				
3	468	0.100	1	464	2	1			
4	363	0.072	3	7	5	1	6	20	321
5	545	0.123	542	1	2				
6	1030	0.033	832	36	73	18	4	65	2
7	661	0.110	1	0	660				
8	914	0.046	52	514	334	8	1		5
9	822	0.105	822						
10	771	0.105	1	1		769			
11	641	0.052	2		639				
12	447	0.091	89	30	139	11	110	60	8
13	250	0.105	244		5	1			
14	545	0.138	540		5				
15	360	0.168	2	355	3				

$\mathcal{I}_2$— RB + Refinement (Entropy=0.168)

Figure 5. The cluster-class distribution of the clustering solutions for the $\mathcal{I}_2$ criterion function for the *sports* dataset, for the repeated-bisections solution and the repeated-bisections followed by k-way refinement.

this solution against the one obtained after performing refinement we can see that the size of clusters 6 and 8 (which are among the looser clusters) increased substantially, whereas the size of some of the tighter clusters decreased (e.g., clusters 5, 10, and 14).

6. Concluding remarks

In this paper we studied seven different global criterion functions for clustering large documents datasets. Four of these functions ($\mathcal{I}_1$, $\mathcal{I}_2$, $\mathcal{G}_1$, and $\mathcal{G}_2$) have been previously proposed for document clustering, whereas the remaining three ($\mathcal{E}_1$, $\mathcal{H}_1$, and $\mathcal{H}_2$) were introduced by us. Our study consisted of a detailed experimental evaluation using fifteen different datasets and three different approaches to find the desired clusters, followed by a theoretical analysis of the characteristics of the various criterion functions. Our experiments showed that $\mathcal{I}_1$ performs poorly whereas $\mathcal{I}_2$ leads to reasonably good results that outperform the solutions produced by some recently proposed criterion functions ($\mathcal{G}_1$ and $\mathcal{G}_2$). Our three new criterion functions performed reasonably well, with the $\mathcal{H}_2$ criterion function achieving the best overall results.

Our analysis showed that the performance difference observed by the various criterion functions can be attributed to the extent to which the criterion functions are sensitive to clusters of different degrees of tightness, and the extent to which they can lead to reasonably balanced solutions. Moreover, our analysis was able to identify a key property of the $\mathcal{I}_1$ criterion function that can be useful in clustering noisy datasets, in which many documents are segregated to a separate "garbage" cluster.

The various clustering algorithms and criterion functions described in this paper are available in the CLUTO clustering toolkit that is available online at http://www.cs.umn.edu/~cluto.

Acknowledgments

We would like to thank the anonymous reviewers and Douglas H. Fisher for their valuable comments and suggestions on the presentation of this paper.

Appendix A: Proofs of $\mathcal{I}_1$'s and $\mathcal{I}_2$'s optimal solution properties

Proof (*Proposition* 1)**:** For contradiction, let $A_{opt} = \{S_1, S_2, \ldots, S_k\}$ be an optimal solution and assume that there exists a document d and clusters S_i and S_j such that $d \in S_i$ and $\delta_i - \delta_j < (\mu_i - \mu_j)/2$. Consider the clustering solution $A' = \{S_1, S_2, \ldots, \{S_i - d\}, \ldots, \{S_j + d\}, \ldots, S_k\}$. Let D_i, C_i, and D_j, C_j be the composite and centroid vectors of cluster $S_i - d$ and S_j, respectively. Then,

$$\begin{aligned}
\mathcal{I}_1(A_{opt}) - \mathcal{I}_1(A') &= \frac{\|D_i + d\|^2}{n_i + 1} + \frac{\|D_j\|^2}{n_j} - \left(\frac{\|D_i\|^2}{n_i} + \frac{\|D_j + d\|^2}{n_j + 1}\right) \\
&= \left(\frac{\|D_i + d\|^2}{n_i + 1} - \frac{\|D_i\|^2}{n_i}\right) - \left(\frac{\|D_j + d\|^2}{n_j + 1} - \frac{\|D_j\|^2}{n_j}\right) \\
&= \left(\frac{2n_i d^t D_i + n_i - {D_i}^t D_i}{n_i(n_i + 1)}\right) - \left(\frac{2n_j d^t D_j + n_j - {D_j}^t D_j}{n_j(n_j + 1)}\right) \\
&= \left(\frac{2n_i\delta_i}{n_i + 1} + \frac{1}{n_i + 1} - \frac{n_i\mu_i}{n_i + 1}\right) \\
&\quad - \left(\frac{2n_j\delta_j}{n_j + 1} + \frac{1}{n_j + 1} - \frac{n_j\mu_j}{n_j + 1}\right) \\
&\approx (2\delta_i - 2\delta_j) - (\mu_i - \mu_j),
\end{aligned}$$

when n_i and n_j are sufficiently large. Since $\delta_i - \delta_j < (\mu_i - \mu_j)/2$, we have $\mathcal{I}_1(A_{opt}) - \mathcal{I}_1(A') < 0$, a contradiction. □

Proof (*Proposition* 2)**:** For contradiction, let $A_{opt} = \{S_1, S_2, \ldots, S_k\}$ be an optimal solution and assume that there exists a document d and clusters S_i and S_j such that $d \in S_i$ and $\delta_i/\delta_j < \sqrt{\mu_i/\mu_j}$. Consider the clustering solution $A' = \{S_1, S_2, \ldots, \{S_i - d\}, \ldots, \{S_j + d\}, \ldots, S_k\}$. Let D_i, C_i, and D_j, C_j be the composite and centroid vectors of cluster $S_i - d$ and S_j, respectively. Then,

$$\begin{aligned}
\mathcal{I}_2(A_{opt}) - \mathcal{I}_2(A') &= \|D_i + d\| + \|D_j\| - (\|D_i\| + \|D_j + d\|) \\
&= \left(\sqrt{{D_i}^t D_i + 1 + 2d^t D_i} - \sqrt{{D_i}^t D_i}\right) \\
&\quad - \left(\sqrt{{D_j}^t D_j + 1 + 2d^t D_j} - \sqrt{{D_j}^t D_j}\right). \qquad (12)
\end{aligned}$$

Now, if n_i and n_j are sufficiently large we have that ${D_i}^t D_i + 2d^t D_i \gg 1$, and thus

$${D_i}^t D_i + 1 + 2d^t D_i \approx {D_i}^t D_i + 2d^t D_i. \qquad (13)$$

Furthermore, we have that

$$\left(\sqrt{D_i{}^t D_i} + \frac{d^t D_i}{\sqrt{D_i{}^t D_i}}\right)^2 = D_i{}^t D_i + \frac{(d^t D_i)^2}{D_i{}^t D_i} + 2d^t D_i \approx D_i{}^t D_i + 2d^t D_i, \qquad (14)$$

as long as $\delta_i^2/\mu_i = o(1)$. This condition is fairly mild as it essentially requires that μ_i is sufficiently large relative to δ_i^2, which is always true for sets of documents that form clusters. Now, using Eqs. (13) and (14) for both clusters, Eq. (12) can be rewritten as

$$\mathcal{I}_2(A_{opt}) - \mathcal{I}_2(A') = \frac{d^t D_i}{\sqrt{D_i{}^t D_i}} - \frac{d^t D_j}{\sqrt{D_j{}^t D_j}} = \frac{\delta_i}{\sqrt{\mu_i}} - \frac{\delta_j}{\sqrt{\mu_j}}.$$

Since $\delta_i/\delta_j < \sqrt{\mu_i/\mu_j}$, we have $\mathcal{I}_2(A_{opt}) - \mathcal{I}_2(A') < 0$, a contradiction. □

References

Available at http://www.cs.umn.edu/~karypis/cluto/files/datasets.tar.gz.

Available from ftp://ftp.cs.cornell.edu/pub/smart.

Beeferman, D., & Berger, A. (2000). Agglomerative clustering of a search engine query log. In *Proc. of the Sixth ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining* (pp. 407–416).

Berry, M., Dumais, S., & O'Brien, G. (1995). Using linear algebra for intelligent information retrieval. *SIAM Review, 37*, 573–595.

Boley, D. (1998). Principal direction divisive partitioning. *Data Mining and Knowledge Discovery, 2:4*.

Cheeseman, P., & Stutz, J. (1996). Baysian classification (AutoClass): Theory and results. In U. Fayyad, G. Piatetsky-Shapiro, P. Smith, & R. Uthurusamy (Eds.), *Advances in Knowledge Discovery and Data Mining* (pp. 153–180). AAAI/MIT Press.

Cheng, C.-K., & Wei, Y.-C. A. (1991). An improved two-way partitioning algorithm with stable performance. *IEEE Transactions on Computer Aided Design, 10:12*, 1502–1511.

Cutting, D., Pedersen, J., Karger, D., & Tukey, J. (1992). Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the ACM SIGIR*. (pp. 318–329). Copenhagen.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39.

Devore, J., & Peck, R. (1997). *Statistics: The exploration and analysis of data*. Belmont, CA: Duxbury Press.

Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *Knowledge Discovery and Data Mining* (pp. 269–274).

Dhillon, I. S., & Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning, 42:1/2*, 143–175.

Ding, C., He, X., Zha, H., Gu, M., & Simon, H. (2001). *Spectral min-max cut for graph partitioning and data clustering*. Technical Report TR-2001-XX, Lawrence Berkeley National Laboratory, University of California, Berkeley, CA.

Duda, R., Hart, P., & Stork, D. (2001). *Pattern classification*. John Wiley & Sons.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of the Second Int'l Conference on Knowledge Discovery and Data Mining*. Portland: OR.

Fisher, D. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*, 139–172.

Fisher, D. (1996). Iterative optimization and simplification of hierarchical clusterings. *Journal of Artificial Intelligence Research, 4*, 147–180.

Guha, S., Rastogi, R., & Shim, K. (1998). CURE: An efficient clustering algorithm for large databases. In *Proc. of 1998 ACM-SIGMOD Int. Conf. on Management of Data*.

Guha, S., Rastogi, R., & Shim, K. (1999). ROCK: A robust clustering algorithm for categorical attributes. In *Proc. of the 15th Int'l Conf. on Data Eng.*

Hagen, L., & Kahng, A. (1991). Fast spectral methods for ratio cut partitioning and clustering. In *Proceedings of IEEE International Conference on Computer Aided Design* (pp. 10–13).

Han, E., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., & Moore, J. (1998). WebACE: A web agent for document categorization and exploartion. In *Proc. of the 2nd International Conference on Autonomous Agents*.

Han, J., Kamber, M., & Tung, A. K. H. (2001). Spatial clustering methods in data mining: A survey. In H. Miller, & J. Han (Eds.), *Geographic data mining and knowledge discovery*. Taylor and Francis.

Hersh, W., Buckley, C., Leone, T., & Hickam, D. (1994). OHSUMED: An interactive retrieval evaluation and new large test collection for research. In *SIGIR-94* (pp. 192–201).

Jackson, J. E. (1991). *A User's guide to principal components*. John Wiley & Sons.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys, 31:3*, 264–323.

Karypis, G., & Han, E. (2000). *Concept indexing: A fast dimensionality reduction algorithm with applications to document retrieval & categorization*. Technical Report TR-00-016, Department of Computer Science, University of Minnesota, Minneapolis. Available on the WWW at URL *http://www.cs.umn.edu/˜karypis*.

Karypis, G., Han, E., & Kumar, V. (1999a). Chameleon: A hierarchical clustering algorithm using dynamic modeling. *IEEE Computer, 32:8*, 68–75.

Karypis, G., Han, E., & Kumar, V. (1999b). Multilevel refinement for hierarchical clustering. Technical Report TR-99-020, Department of Computer Science, University of Minnesota, Minneapolis.

King, B. (1967). Step-wise clustering procedures. *Journal of the American Statistical Association, 69,* 86–101.

Kolda, T., & Hendrickson, B. (2000). Partitioning sparse rectangular and structurally nonsymmetric matrices for parallel computation. *SIAM Journal on Scientific Computing, 21:6*, 2048–2072.

Larsen, B., & Aone, C. (1999). Fast and effective text mining using linear-time document clustering. In *Proc. of the Fifth ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining* (pp. 16–22).

Lewis, D. D. (1999). Reuters-21578 text categorization test collection Distribution 1.0. http://www.research.att.com/∼lewis.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proc. 5th Symp. Math. Statist, Prob* (pp. 281–297).

Meila, M., & Heckerman, D. (2001). An experimental comparison of model-based clustering methods. *Machine Learning, 42*, 9–29.

Ng, R., & Han, J. (1994). Efficient and effective clustering method for spatial data mining. In *Proc. of the 20th VLDB Conference* (pp. 144–155). Santiago, Chile.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14:3*, 130–137.

Puzicha, J., Hofmann, T., & Buhmann, J. M. (2000). A theory of proximity based clustering: Structure detection by optimization. *PATREC: Pattern Recognition*. Pergamon Press. (vol. 33, pp. 617–634).

Salton, G. (1989). *Automatic text processing: The transformation, analysis, & retrieval of information by computer*. Addison-Wesley.

Savaresi, S., & Boley, D. (2001). On the performance of bisecting K-means and PDDP. In *First SIAM International Conference on Data Mining (SDM'2001)*.

Savaresi, S., Boley, D., Bittanti, S., & Gazzaniga, G. (2002) Choosing the cluster to split in bisecting divisive clustering algorithms. In *Second SIAM International Conference on Data Mining (SDM'2002)*.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22:8*, 888–905.

Sneath, P. H., & Sokal, R. R. (1973). *Numerical taxonomy*. London, UK: Freeman.

Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. *KDD Workshop on Text Mining*.

Strehl, A., & Ghosh, J. (2000). Scalable approach to balanced, high-dimensional clustering of market-baskets. In *Proceedings of HiPC*.

TREC (1999). Text REtrieval conference. http://trec.nist.gov.

Zahn, K. (1971). Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computers, C-20*, 68–86.

Zha, H., He, X., Ding, C., Simon, H., & Gu, M. (2001a). Bipartite graph partitioning and data clustering. *CIKM*.
Zha, H., He, X., Ding, C., Simon, H., & Gu, M. (2001b). *Spectral relaxation for K-means clustering*. Technical Report TR-2001-XX, Pennsylvania State University, University Park, PA.
Zhao, Y., & Karypis, G. (2001). *Criterion functions for document clustering: Experiments and analysis*. Technical Report TR #01–40, Department of Computer Science, University of Minnesota, Minneapolis, MN. Available on the WWW at *http://cs.umn.edu/~karypis/publications*.

Received February 21, 2002
Revised May 7, 2003
Accepted May 7, 2003
Final manuscript July 1, 2003