Matrix-based algorithms for document clustering

S. Oliveira and S.-C. Seok

15th March 2005

The clustering problem is the task of assigning each document in a collection to clusters of similar documents. The clustering process does not begin with pre-specified categories; rather it is the purpose of the clustering algorithm to discover natural categories in the collection of documents that it processes. We assume that the initial data for the clustering algorithms consists of a similarity matrix $S = [\, s_{ij} \mid i,\, j \in \mathcal{D}\,]$ where $\mathcal{D}$ is the set of documents and s_{ij} is a measure of the similarity of documents i and j. Typically this matrix is obtained from a transformed word–document matrix W by $S := W^T W$.

A variety of methods have been developed for clustering problems, such as the k-means method and its variants [4, 6]. These methods produce answers that are typically very dependent on the initial data, or on the order in which documents are "seen" by the algorithm. A recent method that has been proposed is based on the computation of eigenvectors which is called the MinMaxCut algorithm proposed by Ding et al. [3], which does not have this defect, which is also very accurate in clustering experiments [2, 3]. However, these algorithms tend to become very expensive. In this abstract we develop an algorithm that retains and even improves on the accuracy of their method, while substantially reducing its cost.

Clustering tasks bear a strong resemblance to graph partitioning problems [7, 1, 5], and similar matrix eigenvalue/eigenvector algorithms can be used for graph partitioning problems. Since the quality of a cluster (the ratio of correctly assigned documents to the total number of documents) is the main objective in clustering, but not in graph partitioning, more stringent algorithms are needed to obtain good results for clustering than for graph partitioning. Nevertheless, some ideas developed for graph partitioning can be used for clustering algorithms.

The two-way MinMaxCut aims to approximately minimize $J_{MMC}(A, B) := s(A, B)/s(A, A) + s(A, B)/s(B, B)$ where $s(X, Y) = \sum_{i \in X,\, j \in Y} s_{ij}$ and $\{A,\, B\}$ is the partitioning $\mathcal{D}$. The continuous relaxation is equivalent to the finding the second smallest eigenvalue λ of $(D - S)q = \lambda D q$, with $D = \mathrm{diag}(d_1, d_2, \cdots, d_n)$ and $d_i = \sum_j s_{ij}$.

The new algorithm that we have developed uses a hierarchical aggregation scheme to create a collection of successively coarser approximations to the original clustering problem. At the coarsest level the continuous relaxation of the original objective function is minimized by means of an eigenvalue/eigenvector

problem. Since the coarsest problem has a greatly reduced number of nodes compared to the original clustering problem, the cost of solving the eigenvalue/eigenvector problem is made much less significant.

Simply taking the clusters on the coarsest level to create the clusters on the original level results in significantly poorer clusters. So after each time the algorithm goes from a coarser to then next finer level, the coarse level nodes in each cluster are disaggregated, but then we also use a local refinement strategy similar to the Kernighan–Lin algorithm to improve accuracy using the given objective function as a measure of quality.

The computational results for this algorithm are very encouraging.

References

[1] S. Barnard and H. Simon. A fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems. In *Proceedings of the Sixth SIAM Conference on Parallel Processing for Scientific Computing*, Norfolk, Virginia, 1993. SIAM, SIAM.

[2] C. Ding, X. He, H. Zha, M. Gu, and H. Simon. A min-max cut algorithm for graph partitioning data clustering. In *ICDM 2001, Proceedings IEEE International Conference on Data Mining, 2001*, pages 107–114. IEEE, 2001.

[3] C. Ding, X. He, H. Zha, M. Gu, and H. Simon. A minmaxcut spectral method for data clustering and graph partitioning. Technical Report 54111, LBNL, December 2003.

[4] V. Faber. Clustering and the continuous k-means algorithm. *Los Alamos Science*, 22:138–144, 1994.

[5] B. Hendrickson and T. G. Kolda. Graph partitioning models for parallel computing. *Parallel Comput.*, 26(12):1519–1534, 2000.

[6] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proc. Fifth Berkeley Symp. Math. Statistics and Probability*, 1:281–296, 1967.

[7] Alex Pothen, Horst D. Simon, and Kang-Pu Liou. Partitioning sparsematrices with eigenvectors of graphs. *SIAM J. Matrix Anal. Appl.*, 11(3):430–452, 1990.