A General Framework for Fast Co-clustering on Large Datasets Using Matrix Decomposition

Abstract

Simultaneously clustering columns and rows (co-clustering) of large data matrix is an important problem with wide applications, such as document mining, microarray analysis, and recommendation systems. Several co-clustering algorithms have been shown effective in discovering hidden clustering structures in the data matrix. For a data matrix of m rows and n columns, the time complexity of these methods is usually in the order of m×n (if not higher). This limits their applicability to data matrices involving a large number of columns and rows. Moreover, an implicit assumption made by existing co-clustering methods is that the whole data matrix needs to be held in the main memory. In this project, we propose a general framework, CRD, for co-clustering large datasets utilizing recently developed sampling-based matrix decomposition methods. The time complexity of our approach is linear in m and n. And it does not require the whole data matrix be in the main memory. Experimental results show that CRD achieves competitive accuracy to existing co-clustering methods but with much less computational cost.

Introduction

Clustering is a fundamental data mining problem with a wide variety of applications. Recently there has been a growing research interest in developing co-clustering algorithms that simultaneously cluster both columns and rows of the data matrix. It has successful applications in gene expression data analysis and text mining.

Although theoretically well studied and widely applied, existing co-clustering algorithms usually have the time complexity in the order of m × n. Such high time complexity limits the applicability of existing algorithms to these large datasets. Furthermore, these algorithms implicitly make the assumption that the whole data matrix is held in the main memory, since the original data matrix needs to be accessed constantly during the execution of the algorithms.

To address these limitations of existing work, in this project, we propose a general co-clustering framework, CRD, for large datasets. This framework is based on recently developed sampling-based matrix decomposition method CUR. Unlike the previous algorithms, the complexity of CRD algorithms is linear in m and n. Moreover, most of the operations in CRD involve only the sampled columns and rows. Therefore, we do not require the whole data matrix be in main memory. This is crucial for large datasets. CRD can be implemented using different algorithms such as k-means or information-theoretic co-clustering methods. We conduct extensive experiments on both synthetic and several well-known real-life datasets.

The CRD Framework

Our *CRD* framework consists of two components.

1. Low rank matrix decomposition: The data matrix is decomposed using a subset of its rows and columns. The decomposition procedure must be fast and accurate.
2. Co-clustering on the subset of rows and columns: The selected rows and columns are co-clustered first. The cluster labels for the rest rows and columns are assigned based on those selected ones. In general, any coclustering algorithm that optimizes its objective function by alternating the clustering of rows and columns can be used in *CRD*.
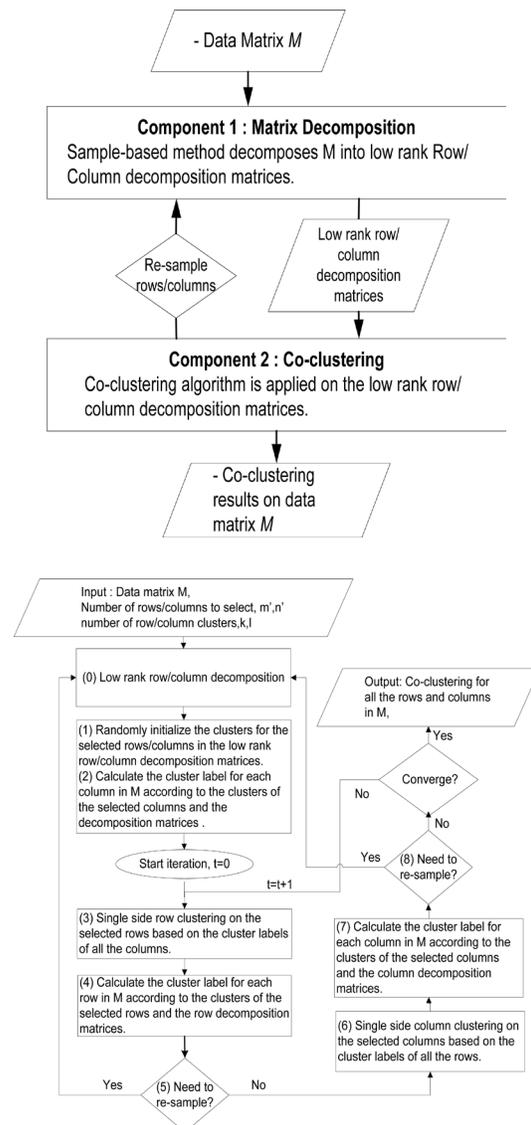
Based on the co-clustering results, a subset of the rows/columns may be returned to the decomposition component for re-sampling. The updated decomposition matrices will be sent back to the co-clustering component to be co-clustered again.

Experiment: 20 Newsgroup Data

CRD-ITC: CRD using information-theoretic co-clustering
CRD-kmeans: CRD using Euclidian distance (k-means) co-clustering
ITC: the original information-theoretic co-clustering algorithm.
Kmeans: the original Euclidian distance co-clustering algorithm
ONMF: the orthogonal nonnegative matrix tri-factorization co-clustering algorithm.

Conclusion

In this project, we proposed a general framework for fast co-clustering on large data, *CRD*. *CRD* has two components. It first decomposes the data matrix into low rank row/column approximation matrices. Then co-clustering algorithms using iterative single-side clustering are used to cluster the approximation matrices. Because of the small size of the approximation matrices, CRD has runtime complexity equal to $O(t(km'n+ln'm+m'm+n'n))$ which is orders of magnitude faster than $O(t(k + l)mn)$, the runtime complexity of the previous co-clustering algorithms.

Current Project Members

Wei Wang, faculty member
Feng Pan, graduate research assistant
Xiang Zhang, graduate research assistant

For More Information

Wei Wang
Professor
919-962-1744
weiwang@cs.unc.edu