SPECTRAL RELAXATION MODELS AND STRUCTURE ANALYSIS FOR K-WAY GRAPH CLUSTERING AND BI-CLUSTERING

M. GU, H. ZHA, C. DING, X. HE, H. SIMON, J. XIA

Abstract. In this paper we consider k-way graph clustering and k-way bipartite graph clustering. Many data types arising from data mining applications can be modeled as graphs and bipartite graphs. Examples include webpage links on the world-wide web, terms and documents in a text corpus, customers and purchasing items in market basket analysis and reviewers and movies in a movie recommender system. In this paper, we discuss models for k-way irregular graph partitioning and k-way bipartite irregular graph partitioning, and discuss the algebraic structures in the eigenvector and singular vector matrices that correspond to the optimal partition. Our discussion contributes to the theoretical understanding of spectral methods for graph partitioning and data mining, which has been a focus of some recent research.

Key words. data mining, data clustering, eigenvalue, perturbation, structure analysis.

1. Introduction. Cluster analysis is an important tool for exploratory data mining applications arising from many diverse disciplines. Informally, cluster analysis seeks to partition a given data set into compact clusters so that data objects within a cluster are more similar than those in distinct clusters. The literature on cluster analysis is enormous including contributions from many research communities. Many traditional clustering algorithms make the assumption that the given data set consists of covariate information (or attributes) for each individual data objects, and clustering can be done within the context of grouping a set of n-dimensional vectors.

In this paper, we concentrate on graph theoretical models for document clustering, where each document is a node in the graph. The weight on each edge (i, j) is chosen to reflect similarities between documents i and j. Clustering in this approach is based on the intuitive notion that documents that are more similar to each other should have a better chance of belonging to the same clusters. We also consider bipartite graph models where terms and documents are modeled as vertices in a bipartite graph with edges of the graph indicating the co-occurrence of terms and documents, and edge weights indicating the frequency of this co-occurrence. Clustering in this latter approach is based on the intuitive notion that documents in the same topic tend to use the same set of terms while the set of terms that can be used to describe a particular topic tends to appear in documents pertaining to that topic.

Graph partitioning has very broad range of applications. For example, the mesh of a 2D surface of an airfoil or a 3D engine cylinder can be modeled as a nearly-regular graph. Partitioning such a mesh into subdomains for distributed memory processors is a common task with well-developed software such as that in [14, 12]. Page linkages on the World Wide Web form another kind of graph, which are are highly irregular or random, with dramatically varying node degrees. Partitioning such graphs is useful to automatically identify topics from the retrieved webpages for a user query [6, 7, 11]. In this paper we emphasize graph partition as data clustering using a graph model.

Some of the most effective tools for clustering analysis have been spectral methods, where one computes some eigenvalues and eigenvectors of a Laplacian matrix (or singular values and singular vectors of certain matrix) related to the given graph and constructs data clusters based on such spectral information [1, 5, 8, 13, 10, 17, 18, 21, 24]. While such methods have existed and been applied in a number of fields for many years, they are heuristic in nature due to the high complexity of the data clustering problem.

Recently, there have been some interesting research towards a theoretical understanding of spectral methods. In particular, Papadimitriou, Raghavan, Tamaki and Vempala [19] showed that for documents generated from a probabilistic model that they developed, document clusters on each topic, with high probability, are determined by the subspaces produced by Latent Semantic Indexing (LSI). Azar, Fiat, Karlin, McSherry and Saia [2] further generalize the results of [19] to show that LSI actually works in a much wider variety of settings. Ding, He and Zha [6, 7] developed methods for web link and text data graph partitioning based on spectral perturbation theory.

With approaches similar to those of [6, 7, 2], in this paper, we show that important algebraic structures in general exist in the eigenvector and singular vector matrices for data clusters for both graph clustering and bi-clustering. We also present experimental results in text data clustering that confirm the existence of such algebraic structures.

The rest of the paper is organized as follows: in Section 2 we review normalized cut for k-way clustering. In Section 3 we propose a new criterion for graph partitioning

which tends to produce balanced clusters and show that our criterion can be related to a partial singular value decomposition of the weight matrix of the bipartite graph. In Section 4, we perform a spectral perturbation analysis to reveal important algebraic structures hidden in the partial singular value decomposition. In Section 5 we propose a new criterion for bipartite graph partitioning and perform a similar spectral analysis. In section 6, we describe experimental results that confirm the existence of such structures in bi-clustering a dataset of newsgroup articles.

2. Normalized Cut for k-way clustering. We denote a weighted undirected graph by $G(V, E, W)$ where V is the vertex set and E is the edge set, and $W = (w_{ij})$ is the weight matrix with $w_{ij} \geq 0$ denoting the edge weight if there is an edge between vertex i and j, i.e., $\{i, j\} \in E$, otherwise we set $w_{ij} = 0$. Let $\Pi = \{V_i\}_{i=1}^k$ be a k-way disjoint partition of V such that

$$V = \cup_{i=1}^k V_i. \tag{1}$$

The k-way Normalized Cut problem is to find a partition Π that minimizes the objective function

$$\text{Ncut}\,(\Pi) = \frac{W(V_1, V_1^c)}{W(V_1, V)} + \frac{W(V_2, V_2^c)}{W(V_2, V)} + \cdots + \frac{W(V_k, V_k^c)}{W(V_k, V)}. \tag{2}$$

The normalized cut formulation has been discussed in [4] for graph partitioning in general and in [21] for image segmentation in particular. Ding *el al* [7] discussed normalized cut for data clustering. For a fixed clustering of the vertices of G, reorder the rows and columns of W accordingly such that

$$W = \begin{bmatrix} W_{11} & W_{12} & \cdots & W_{1k} \\ W_{21} & W_{22} & \cdots & W_{2k} \\ \vdots & \vdots & & \vdots \\ W_{k1} & W_{k2} & \cdots & W_{kk} \end{bmatrix},$$

and the rows of W_{i1} correspond to the vertices in V_i, $i = 1, \ldots, k$. Let e be the vector with all its elements equal to 1. Let $D = \text{diag}(D_1, \ldots, D_k)$ be the diagonal matrix such that $We = De$. For any m-by-n matrix $H = (h_{ij})$, denote

$$\text{s}(H) = \sum_{i=1}^{m} \sum_{j=1}^{n} h_{ij},$$

i.e., $\text{s}(H)$ is the sum of all the elements of H. It is easy to verify that

$$W(V_i, V) = \text{s}(D_i) \quad \text{and} \quad W(V_i, V_i^c) = \text{s}\left(\sum_{j \neq i} W_{ij}\right) = \text{s}(D_i) - \text{s}(W_{ii}). \tag{3}$$

Let x_i be a vector accordingly partitioned with that of W with all elements equal to zero except those corresponding to rows of W_{ii}, i.e.,

$$x_i = [0 \cdots 0 \;\; 1 \cdots 1 \;\; 0 \cdots 0]^T.$$

It follows from (3) that

$$W(V_i, V_i^c) = x_i^T (D - W) x_i, \quad W(V_i, V) = x_i^T D x_i. \tag{4}$$

Therefore,

$$\begin{aligned}\mathrm{Ncut}\,(\Pi) &= \frac{x_1^T(D-W)x_1}{x_1^T D x_1} + \frac{x_2^T(D-W)x_2}{x_2^T D x_2} + \cdots + \frac{x_k^T(D-W)x_k}{x_k^T D x_k} \\ &= k - \left(\frac{x_1^T\, W\, x_1}{x_1^T\, D\, x_1} + \frac{x_2^T\, W\, x_2}{x_2^T\, D\, x_2} + \cdots + \frac{x_k^T\, W\, x_k}{x_k^T\, D\, x_k} \right). \qquad (5)\end{aligned}$$

Let $y_i = D^{1/2}x_i/\|D^{1/2}x_i\|_2$ and $Y_k = [y_1, y_2, \ldots, y_k]$. Then $Y_k^T\, Y_k = I_k$ and

$$\mathrm{Ncut}\,(\Pi) = \mathrm{trace}(Y_k^T D^{-1/2}(D-W)D^{-1/2}Y_k) = k - \mathrm{trace}(Y_k^T \widehat{W} Y_k), \qquad (6)$$

where

$$\widehat{W} = D^{-1/2}\, W\, D^{-1/2}.$$

The largest eigenvalue of $\widehat{W}$ is always 1, with corresponding eigenvector $D^{1/2}\, e$.

If we temporarily ignore the fact that the elements of x_i are either zero or one, and allow them to take continuous values, we arrive at the following relaxed optimization problem

$$\min \mathrm{Ncut}\,(\Pi) \geq k - \max_{Y_k^T\, Y_k = I_k} \mathrm{trace}\left(Y_k^T\, \widehat{W}\, Y_k\right).$$

It is known that the maximum on the right hand side is achieved when Y_k is taken to be any orthonormal basis for the subspace spanned by the eigenvectors pertaining to the k *largest* eigenvalues of $\widehat{W}$. Hence we have a lower bound on $\mathrm{Ncut}\,(\Pi)$ as (cf. [3])

$$\min_{\Pi} \mathrm{Ncut}\,(\Pi) \geq k - \sum_{j=1}^{k} \widehat{\lambda}_j, \qquad (7)$$

where $\widehat{\lambda}_1, \cdots, \widehat{\lambda}_k$ are the largest k eigenvalues of $\widehat{W}$.

3. k-Way Min-Max Cut. In the normalized cut (2), each cut $W(V_i, V_i^c)$ is weighted against the association between V_i and the whole graph V. One could argue that it might be more meaningful to weigh each cut $W(V_i, V_i^c)$ against the association within V_i. This distinction becomes more pronounced when the cutsizes $W(V_i, V_i^c)$ stops beeing tiny, as is the case for most practical data clustering problems. In the following we consider such a weighting scheme, formulated as the following Min-max cut problem:

$$\mathrm{Mcut}\,(\Pi) = \frac{W(V_1, V_1^c)}{W(V_1, V_1)} + \frac{W(V_2, V_2^c)}{W(V_2, V_2)} + \cdots + \frac{W(V_k, V_k^c)}{W(V_k, V_k)}, \qquad (8)$$

where as before $W(V_i, V_i^c) = x_i^T\, (D-W)\, x_i$. For $k = 2$, this formulation was discussed in [7]. The Min-max Cut tries to minimize inter-cluster similarity while maximizing intra-cluster similarity. We will discuss this more when we discuss the solution to (8).

The numerators in the above equation can be rewritten as (4), and the denominators can be rewritten as $W(V_i, V_i) = x_i^T\, W\, x_i$. Therefore,

$$\begin{aligned}\mathrm{Mcut}\,(\Pi) &= \frac{x_1^T\,(D-W)\,x_1}{x_1^T\, W\, x_1} + \frac{x_2^T\,(D-W)\,x_2}{x_2^T\, W\, x_2} + \cdots + \frac{x_k^T\,(D-W)\,x_k}{x_k^T\, W\, x_k} \\ &= \frac{x_1^T\, D\, x_1}{x_1^T\, W\, x_1} + \frac{x_2^T\, D\, x_2}{x_2^T\, W\, x_2} + \cdots + \frac{x_k^T\, D\, x_k}{x_k^T\, W\, x_k} - k. \qquad (9)\end{aligned}$$

Define y_i, Y_k and $\widehat{W}$ as before, we have

$$\text{Mcut}\,(\Pi) = \frac{1}{y_1^T\,\widehat{W}\,y_1} + \frac{1}{y_2^T\,\widehat{W}\,y_2} + \cdots + \frac{1}{y_k^T\,\widehat{W}\,y_k} - k. \tag{10}$$

We would like to find the right permutation Π that minimizes $\text{Mcut}\,(\Pi)$.

To relax this problem into a meaningful continuous problem, we note that the matrix $\widehat{W}$, while symmetric, is usually not (semi-)positive definite. Consequently, it is possible to choose y_i to make $y_i^T\,\widehat{W}\,y_i < 0$ and yet extremely close to 0. In other words, the minimum value of the right hand side in (10) would be $-\infty$ if the only constraint on y_i is that $Y_k^T\,Y_k = I_k$. To avoid this problem, we relax the problem (9) into that of minimizing the right hand side of (10) under the conditions that

$$Y_k^T\,Y_k = I_k, \quad \text{and} \quad y_i^T\,\widehat{W}\,y_i > 0, \;\; i = 1, \cdots, k. \tag{11}$$

Conditions (11) ensure that the relaxed minimization problem has a positive minimum value. The following theorem provides the solution to this relaxed problem.

THEOREM 3.1. *Let*

$$\widehat{\lambda}_1 \geq \widehat{\lambda}_2 \geq \cdots \geq \widehat{\lambda}_n$$

be the eigenvalues of $\widehat{W}$ *and assume that* $\widehat{\lambda}_1 + \widehat{\lambda}_2 + \cdots + \widehat{\lambda}_k > 0$. *Then*

$$\min_{Y_k^T\,Y_k = I_k,\, y_i^T\,\widehat{W}\,y_i > 0} \sum_{i=1}^{k} \frac{1}{y_i^T\,\widehat{W}\,y_i} = \frac{k^2}{\sum_{i=1}^{k}\widehat{\lambda}_i}. \tag{12}$$

In addition, this minimum is achieved by any orthonormal basis $\{y_1, \cdots, y_k\}$ *of the subspace spanned by the eigenvectors pertaining to the largest* k *eigenvalues of* $\widehat{W}$ *which further satisfies*

$$y_i^T\,\widehat{W}\,y_i = \frac{\sum_{i=1}^{k}\widehat{\lambda}_i}{k}.$$

Proof. Let $\widetilde{Y}$ be the solution to the minimization problem (12). Then there exists an orthonormal matrix $\widetilde{Y}^c$ such that the matrix $(\widetilde{Y}\;\widetilde{Y}^c)$ is orthogonal. Define

$$\widetilde{W} = (\widetilde{Y}\;\widetilde{Y}^c)^T\,\widehat{W}\,(\widetilde{Y}\;\widetilde{Y}^c) = \begin{pmatrix} \widetilde{W}_{11} & \widetilde{W}_{12} \\ \widetilde{W}_{21} & \widetilde{W}_{22} \end{pmatrix}, \quad \text{and} \quad \mathcal{L}(Y_k) = \sum_{i=1}^{k} \frac{1}{y_i^T\,\widetilde{W}\,y_i}, \tag{13}$$

with $Y_k = (y_1, \cdots, y_k)$ satisfying $Y_k^T\,Y_k = I_k$. It is clear from the way $\widetilde{W}$ is defined that

$$Y^* = \begin{pmatrix} I_k \\ 0 \end{pmatrix} = (e_1, \cdots, e_k)$$

consists the optimal solution vectors that solve the minimization problem on the left hand side of (12) where the matrix $\widehat{W}$ is replaced by $\widetilde{W}$. For convenience, we will assume that we have re-arranged all the ratios so that the sequence $\{e_i^T\,\widetilde{W}\,e_i\}$ is in decreasing order.

Let $\delta Y^{(2)} \in \mathcal{R}^{(n-k)\times k}$ be any sufficiently small perturbation. It is easy to show that there exists a perturbation

$$\delta Y^{(1)} = O\left(\left\|\delta Y^{(2)}\right\|_2^2\right) \in \mathcal{R}^{k\times k} \tag{14}$$

such that the perturbation

$$\delta Y = \begin{pmatrix} \delta Y^{(1)} \\ \delta Y^{(2)} \end{pmatrix} = (\delta y_1, \cdots, \delta y_k)$$

exactly satisfies the equation

$$(Y^* + \delta Y)^T \ (Y^* + \delta Y) = I_k. \tag{15}$$

In other words, $Y^* + \delta Y$ exactly satisfies the orthogonality constraint and hence is a feasible solution. Some algebra reveals that:

$$\begin{aligned} \mathcal{L}(Y^* + \delta Y) - \mathcal{L}(Y^*) &= -2\sum_{i=1}^{k} \frac{\delta y_i^T \ \widetilde{W}\, e_i}{\left(e_i \ \widetilde{W}\, e_i\right)^2} + O(\|\delta Y\|^2) \\ &= -2 \text{ trace } \left(\delta Y^T \ \widetilde{W} \ \left(\text{diag}(e_1^T \ \widetilde{W}\, e_1, \cdots, e_k^T \ \widetilde{W}\, e_k)\right)^{-2}\right). \end{aligned} \tag{16}$$

Due to (14), linear perturbation terms in (16) that involve $\delta Y^{(1)}$ are second order perturbation terms in $\delta Y^{(2)}$. For $\mathcal{L}(Y^*)$ to be a local minimum, all the linear terms involving $\delta Y^{(2)}$ on the right hand side of (16) must vanish. This can happen if and only if partition (13) satisfies $\widetilde{W}_{12} = 0$ and $\widetilde{W}_{21} = 0$.

It follows that eigenvalues of $\widetilde{W}_{11}$ must be those of $\widetilde{W}$, and hence of $\widehat{W}$. Let these eigenvalues be $\widetilde{\lambda}_1, \cdots, \widetilde{\lambda}_k$, and let the diagonal entries of $\widetilde{W}_{11}$ be $\widetilde{w}_1, \cdots, \widetilde{w}_k$. It is well known that

$$\mathcal{L}\,(Y^*) = \sum_{i=1}^{k} \frac{1}{e_i^T \ \widetilde{W}\, e_i} = \sum_{i=1}^{k} \frac{1}{\widetilde{w}_i} \geq \frac{k^2}{\sum_{i=1}^k \widetilde{w}_i} = \frac{k^2}{\text{trace}\left(\widetilde{W}_{11}\right)} = \frac{k^2}{\sum_{i=1}^k \widetilde{\lambda}_i},$$

where the inequality becomes equality if and only if

$$\widetilde{w}_j = \frac{\sum_{i=1}^k \widetilde{\lambda}_i}{k} \quad \text{for all } j = 1, \cdots, k.$$

Combining these results, we now have

$$\min_{Y_k^T \ Y_k = I_k, \ y_i^T \ \widehat{W} \ y_i > 0} \sum_{i=1}^{k} \frac{1}{y_i^T \ \widehat{W}\, y_i} \geq \mathcal{L}(Y^*) \geq \frac{k^2}{\sum_{i=1}^k \widetilde{\lambda}_i} \geq \frac{k^2}{\sum_{i=1}^k \widehat{\lambda}_i}.$$

Furthermore, there exists an orthogonal matrix $\widetilde{Q}$ such that (see [25])

$$\widetilde{Q}^T \ \widehat{W} \widetilde{Q} = \begin{pmatrix} \widetilde{W}_1 & \mathbf{0} \\ \mathbf{0} & \widetilde{W}_2 \end{pmatrix},$$

where $\widetilde{W}_1 \in \mathcal{R}^{k\times k}$ is such that every one of its main diagonal entries is equal to $\left(\sum_{i=1}^k \widehat{\lambda}_i\right)/k$. Taking the first k columns of $\widetilde{Q}$ as the y_i vectors, we obtain

$$\min_{Y_k^T Y_k = I_k,\, y_i^T \widehat{W}\, y_i > 0} \sum_{i=1}^k \frac{1}{y_i^T \widehat{W}\, y_i} \le \frac{k^2}{\sum_{i=1}^k \widehat{\lambda}_i}.$$

Since this upper bound is exactly equal to the earlier lower bound, the theorem must be true. □

Since at the optimal solution every term is the same, namely

$$y_i^T \widehat{W}\, y_i = \frac{\widehat{\lambda}_1 + \cdots + \widehat{\lambda}_k}{k},$$

It follows that the following mini-max problem

$$\min_{Y_k^T Y_k = I_k,\, y_i^T \widehat{W}\, y_i > 0} \max_{1\le i\le k} \frac{1}{y_i^T \widehat{W}\, y_i} \tag{17}$$

has exactly the same optimal solution, with a different optimal value $\dfrac{k}{\sum_{i=1}^k \widehat{\lambda}_i}$. Hence the continuous version of the Mcut problem is equivalent to a min-max optimization problem.

Since (10) is a continuous relaxation of (9), the fact that the ratios in (10) are equal at optimal solution suggests that the ratios of (9) should also be rather "close" to each other. For $k = 2$, we observe that the numerators of (9) are identical, hence we can expect denominators in (9) to be rather "close" to each other, which is to say that we should expect $W(V_1)$ and $W(V_2)$ to be rather "close" to each other (see [7] for more detailed discussion). This automatic balancing among the clusters makes Mcut a much-desired objective function for data clustering.

4. Spectral Structure Analysis. Theorem 3.1 states that the solution vectors for the relaxed Mcut problem must lie in the eigenspace spanned by the eigenvectors pertaining to the k largest eigenvalues of $\widehat{W}$. Recently, researchers have developed schemes to compute clustering information by applying variations of the K-means algorithm to these eigenvectors (see for example [5, 8, 13, 21]). It turns out that there is an asymptotic algebraic structure in these eigenvectors that reveals a great deal of information on the individual clusters. The following lemma of Stewart [23] forms the theoretical basis for analyzing this algebraic structure. It was first used in [2] in the analysis of their data mining models. Similar discussions also appeared in [6].

LEMMA 4.1. *Assume that λ is a common eigenvalue of symmetric matrices $A_1, A_2, \cdots, A_k$, with corresponding unit eigenvectors $u_1, u_2, \cdots, u_k$. Further assume that λ is well-separated from all other eigenvalues of these k matrices. Define*

$$A = \begin{pmatrix} A_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & A_k \end{pmatrix} \quad \textit{and} \quad \widetilde{A} = A + \delta A,$$

where δA is a small symmetric perturbation. For the k eigenvalues of $\widetilde{A}$ that are perturbed from λ, denote their corresponding unit eigenvectors as $\widetilde{u}_1, \widetilde{u}_2, \cdots, \widetilde{u}_k$. Then

$$(\widetilde{u}_1\ \cdots\ \widetilde{u}_k) = \begin{pmatrix} u_1\, v_1^T \\ \vdots \\ u_k\, v_k^T \end{pmatrix} + O\left(\|\delta A\|/\|A\|\right), \tag{18}$$

where $(v_1 \cdots v_k)$ *is a* $k \times k$ *orthogonal matrix.*

To see how Lemma 4.1 could be of help in our search for good clusters, we note that $\widehat{W}$ can be rewritten in the following form:

$$\widehat{W} = \begin{pmatrix} \widehat{W}_1 & & \\ & \ddots & \\ & & \widehat{W}_k \end{pmatrix} + \delta W,$$

where $\widehat{W}, \widehat{W}_1, \cdots, \widehat{W}_k$ are the scaled weight matrices for the whole data set and the k clusters in partition Π; and δW is the sum of off-diagonal terms after scaling and all the other terms that make the above equation valid. δW should be reasonably small as long as the cross terms in the partition Π are sufficient small relative to the weights inside each cluster.

All $k+1$ matrices, $\widehat{W}, \widehat{W}_1, \cdots, \widehat{W}_k$, have 1 as their largest eigenvalue by construction. Assuming that the well-separatedness condition holds and that the cutsizes are sufficiently small, Lemma 4.1 suggests that the partition information is built into the algebraic structure (18) of the eigenvectors pertaining to the largest k eigenvalues of $\widehat{W}$.

In Section 6, we experimentally verify the existence of such a structure. Our experiments show that for our target data clustering applications, the exact structure as indicated in Lemma 4.1 does *not* in general exist. This is not surprising since the cutsizes for such applications are anything but tiny. Instead, our experiments indicate that an algebraic structure similar to (18) indeed exists in general for our target applications.

It is interesting to note that equation (18) suggests a link to the k-means strategy used by various authors to uncover k-way partitioning information [5, 21]. In such a strategy, k eigenvectors are computed, and a k-means strategy is applied to the transpose of the corresponding eigenvector matrix. It is easy to show that this strategy leads to an eigenvector structure in the form of (18) with $u_1, u_2, \cdots u_k$ replaced by vectors of all 1's and $(v_1,\ v_2 \cdots v_k)$ an arbitrary set of centroid vectors.

Lemma 4.1 can be slightly modified to hold for any eigenvalue of A, not just the extreme ones or those with multiplicity k. In fact, let λ be any eigenvalue of A. If it is a single eigenvalue of the block A_i with unit eigenvector w_i and is well-separated from all the other eigenvalues of A, then the corresponding perturbed eigenvector for $\widetilde{A}$ is little changed:

$$\begin{pmatrix} 0 \\ \vdots \\ w_i \\ \vdots \\ 0 \end{pmatrix} + +O\left(\|\delta A\|_2/\|A\|_2\right).$$

On the other hand, assume that λ is a common eigenvalue of blocks $A_{i_1}, A_{i_2}, \cdots, A_{i_s}$ for $s \leq k$, with corresponding unit eigenvectors $w_{i_1}, w_{i_2}, \cdots, w_{i_s}$, and that λ is also well-separated from all other eigenvalues of A. Then the s perturbed eigenvectors of $\widetilde{A}$, have the following block-wise structure: for each block A_{i_t}, the eigenvector components are those of $w_{i_t}\ v_{i_t}^T + O\left(\|\delta A\|_2/\|A\|_2\right)$, where $(v_{i_1} \cdots v_{i_s})$ is an $s \times s$ orthogonal matrix; for all other blocks, the corresponding eigenvector components are $O\left(\|\delta A\|_2/\|A\|_2\right)$.

5. k-way Bipartite Graph Partitioning. A graph $G(V;E)$ is bipartite with vertex classes X and Y if $V = X \cup Y$ and each edge in E has one endpoint in X and one endpoint in Y. We consider a weighted bipartite graph $G(X;Y;W^b)$ with $W^b = (w_{ij})$, where w_{ij} denotes the weight of the edge between vertex i and j. In the context of document clustering, X represents the set of terms and Y represents the set of documents, and w_{ij} can be used to denote the number of times term i occurs in document j. We consider A k-way vertex partition Π of $G(X;Y;W^b)$ of the form

$$\begin{aligned} X &= X_1 \cup X_2 \cup \cdots \cup X_k, \\ Y &= Y_1 \cup Y_2 \cup \cdots \cup Y_k. \end{aligned} \tag{19}$$

We say that a pair of vertices $x \in X$ and $y \in Y$ is matched with respect to the partition (19) if $x \in X_j$ and $y \in Y_j$ for some $1 \le j \le k$.

Intuitively, we would like to find partitions of the form (19) where there are as many matches between all X_j and Y_j pairs as possible, and there are as few matches between X_i and Y_j (with $i \neq j$) pairs as possible. This would give rise to partitions with matches concentrated between all X_j and Y_j pairs. In the context of document clustering, this strategy leads to the desired tendency of discovering document clusters $Y_1, \cdots, Y_k$ which find their most used key terms in $X_1, \cdots, X_k$, respectively.

To establish our k-way partition models, we note that treated as a regular graph, the weight matrix W for the bipartite graph $G(X, Y; W^b)$ takes the form

$$W = \begin{pmatrix} \mathbf{0} & W^b \\ (W^b)^T & \mathbf{0} \end{pmatrix}. \tag{20}$$

The zero diagonal blocks in W speak to the fact that we did not consider any pairwise relationship between a term and a document in this vector space model. Hence it is clear that this k-way bi-clustering problem is nothing but a special case of the clustering problem we have discussed in some detail in previous sections. Hence all the theory and numerical schemes apply equally well in this case. In the following, we rewrite the theory and schemes in a form that explicitly takes into account the special structure in W.

In the context of document clustering, the block rows of W correspond to the partitioning of terms and block columns of W the documents. In addition, we define x_j and y_j to be the n- and m-dimensional vectors, respectively. The components of x_j and y_j are all zero except those corresponding to the j-th block rows and columns of W, respectively, which are 1. The partitions (1) and (19) are connected by the relationship

$$V_i = X_i \cup Y_i, \quad i = 1, \cdots, k. \tag{21}$$

5.1. Bipartite Normalized Cut. In light of equations (20) and (21), we rewrite terms in equation (2) for bipartite graph as follows,

$$\begin{aligned} W(V_r, V_r^c) &= \sum_{j \neq r} \mathrm{s}(W_{rj}) + \sum_{i \neq r} \mathrm{s}(W_{ir}) = \sum_{j \neq r} x_r^T\, W^b\, y_j + \sum_{i \neq r} x_i^T\, W^b\, y_r. \\ W(V_r, V) &= \sum_{j=1}^{k} \mathrm{s}(W_{rj}) + \sum_{i=1}^{k} \mathrm{s}(W_{ir}) = \sum_{j=1}^{1} x_r^T\, W^b\, y_j + \sum_{i=1}^{k} x_i^T\, W^b\, y_r. \end{aligned}$$

To further simplify (2) for the bipartite graph, we introduce two diagonal matrices

$$D = \mathrm{diag}(d_1, \cdots, d_k) \quad \text{and} \quad F = \mathrm{diag}(f_1, \cdots, f_k),$$

where d_j and f_j are the sums of the elements in the j-th row and column of W^b, respectively. We assume that the underlining bipartite graph is connected so that $D > 0$ and $F > 0$. Now equation (2) simplifies to

$$(22)\quad \mathrm{Ncut}(\Pi) = \sum_{r=1}^{k} \frac{x_r^T\, D\, x_r + y_r^T\, F\, y_r - 2x_r^T\, W^b\, y_r}{x_r^T\, D\, x_r + y_r^T\, F\, y_r} = k - \sum_{r=1}^{k} \frac{2, x_r^T\, W^b\, y_r}{x_r^T\, D\, x_r + y_r^T\, F\, y_r}.$$

In the spirit of (20), we further let

$$\widehat{z}_r = \begin{pmatrix} \widehat{x}_r \\ \widehat{y}_r \end{pmatrix} = \begin{pmatrix} D^{1/2}\, x_r \\ F^{1/2}\, y_r \end{pmatrix} \Big/ \left\| \begin{matrix} D^{1/2}\, x_r \\ F^{1/2}\, y_r \end{matrix} \right\| \quad \text{and} \quad \mathcal{Z} = \begin{pmatrix} \widehat{x}_1 & \cdots & \widehat{x}_k \\ \widehat{y}_r & \cdots & \widehat{y}_k \end{pmatrix}.$$

It follows that

$$\mathcal{Z}^T\, \mathcal{Z} = I_k \quad \text{and} \quad \frac{2\, x_r^T\, W\, y_r}{x_r^T\, D\, x_r + y_r^T\, F\, y_r} = \begin{pmatrix} \widehat{x}_r \\ \widehat{y}_r \end{pmatrix}^T \begin{pmatrix} 0 & \widehat{W}^b \\ \left(\widehat{W}^b\right)^T & 0 \end{pmatrix} \begin{pmatrix} \widehat{x}_r \\ \widehat{y}_r \end{pmatrix},$$

where $\widehat{W}^b = D^{1/2}\, W^b\, F^{1/2}$. Combining all these relations, we see that the bipartite Normalized Cut (2) can be further simplified to

$$(23)\qquad \mathrm{Ncut}(\Pi) = k - \mathrm{trace}\left(\mathcal{Z}^T \begin{pmatrix} 0 & \widehat{W}^b \\ \left(\widehat{W}^b\right)^T & 0 \end{pmatrix} \mathcal{Z} \right).$$

where $\mathcal{Z}$ is column orthogonal.

Let the largest k singular values of $\widehat{W}^b$ be

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k,$$

and let

$$U_k = (u_1\ u_2 \ldots u_k) \quad \text{and} \quad V_k = (v_1\ v_2 \ldots v_k)$$

be the corresponding left and right singular vectors, respectively. The relaxed problem of minimizing the trace in (23) over all column orthogonal matrices can be solved by setting columns of $\mathcal{Z}$ to be any orthonormal basis for the subspace spanned by columns of the matrix $\begin{pmatrix} U_k \\ V_k \end{pmatrix}$. Bipartite normalized cuts have been discussed in [5, 24].

5.2. Min-Max Cut. We can also explicitly rewrite Min-Max Cut (8) for the bipartite graph alone the lines of Section 5.1. In fact, the denominators of (8) can be rewritten as

$$W(V_r, V_r) = 2\, \mathrm{s}(W_{rr}) = 2\, x_r^T\, W^b\, y_r.$$

and equation (8) becomes

$$\begin{aligned} \mathrm{Mcut}(\Pi) &= \sum_{r=1}^{k} \frac{x_r^T\, D\, x_r + y_r^T\, F\, y_r - 2x_r^T\, W^b\, y_r}{2\, x_r^T\, W^b\, y_r} = \sum_{r=1}^{k} \frac{x_r^T\, D\, x_r + y_r^T\, F\, y_r}{2, x_r^T\, W^b\, y_r} - k \\ &= \sum_{r=1}^{k} \frac{1}{2\, \widehat{x}_r^T\, \widehat{W}^b\, \widehat{y}_r} - k \\ &= \sum_{r=1}^{k} 1 \Big/ \widehat{z}_r^T \begin{pmatrix} 0 & \widehat{W}^b \\ \left(\widehat{W}^b\right) & 0 \end{pmatrix} \widehat{z}_r - k \qquad (24) \end{aligned}$$

For the relaxed continuous problem of minimizing the sum in the last expression, the optimal solution satisfies

$$2\,\widehat{x}_r^T\,\widehat{W}^b\,\widehat{y}_r = \frac{\sum_{r=1}^{k}\sigma_r}{k},$$

and $\{\widehat{z}_r\}$ form an orthonormal basis for the subspace spanned by columns of the matrix $\begin{pmatrix} U_k \\ V_k \end{pmatrix}$.

Similar to the discussion on general matrices, the fact that each ratio in the sum (8) is the same at the optimal solution of the relaxed problem is an indication that one should expect Min-Max Cuts to produce balanced bi-clusters in general, which is a very desired feature.

5.3. Bi-Clustering as a Singular Value Perturbation Problem. The general eigenvalue perturbation results of Section 4 also have their counterparts for bi-clustering bipartite graphs. Lemma 5.1 below first appeared in [2].

LEMMA 5.1. *Let σ be a common singular value of matrices $A_1, A_2, \cdots, A_k$, with corresponding unit right singular vectors $u_1, u_2, \cdots, u_k$ and unit left singular vectors $v_1, v_2, \cdots, v_k$. Further assume that σ is well-separated from all other singular values of all k matrices. Define*

$$A = \begin{pmatrix} A_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & A_k \end{pmatrix} \quad \textit{and} \quad \widetilde{A} = A + \delta A,$$

where δA is a small perturbation. For the k singular values of $\widetilde{A}$ that are perturbed from σ, denote their corresponding unit right $\widetilde{u}_1, \widetilde{u}_2, \cdots, \widetilde{u}_k$ and unit left singularvectors as $\widetilde{v}_1, \widetilde{v}_2, \cdots, \widetilde{v}_k$. Then

$$(\widetilde{u}_1\ \cdots\ \widetilde{u}_k) = \begin{pmatrix} u_1\,w_1^T \\ \vdots \\ u_k\,w_k^T \end{pmatrix} + O\left(\|\delta A\|/\|A\|\right), \tag{25}$$

$$(\widetilde{v}_1\ \cdots\ \widetilde{v}_k) = \begin{pmatrix} v_1\,w_1^T \\ \vdots \\ v_k\,w_k^T \end{pmatrix} + O\left(\|\delta A\|/\|A\|\right), \tag{26}$$

where $(w_1\ \cdots\ w_k)$ is a $k \times k$ orthogonal matrix.

It is interesting to point out that the same set of vectors $w_1 \cdots w_k$ appear in both the perturbed left singular vectors and perturbed right singular vectors.

To put Lemma 5.1 into use for finding good bi-clusters, we rewrite $\widehat{W}^b$ as

$$\widehat{W}^b = \begin{pmatrix} \widehat{W}_1^b & & \\ & \ddots & \\ & & \widehat{W}_k^b \end{pmatrix} + \delta W,$$

where $\widehat{W}^b, \widehat{W}_1^b, \cdots, \widehat{W}_k^b$ are the scaled weight matrices for the whole data set and the k clusters in partition Π; and δW is the sum of off-diagonal terms after scaling and all the other terms that make the above equation valid. δW should be reasonably

TABLE 1
Newsgroups and Their Numbering

NG1	`alt.atheism`	**NG11**	`rec.sport.hockey`
NG2	`comp.graphics`	**NG12**	`sci.crypt`
NG3	`comp.os.ms-windows.misc`	**NG13**	`sci.electronics`
NG4	`comp.sys.ibm.pc.hardware`	**NG14**	`sci.med`
NG5	`comp.sys.mac.hardware`	**NG15**	`sci.space`
NG6	`comp.windows.x`	**NG16**	`soc.religion.christian`
NG7	`misc.forsale`	**NG17**	`talk.politics.guns`
NG8	`rec.autos`	**NG18**	`talk.politics.mideast`
NG9	`rec.motorcycles`	**NG19**	`talk.politics.misc`
NG10	`rec.sport.baseball`	**NG20**	`talk.religion.misc`

small as long as the cross terms in the partition Π are sufficiently small relative to the weights inside each cluster.

All $k+1$ matrices, $\widehat{W}, \widehat{W}_1^b, \cdots, \widehat{W}_k^b$, have 1 as their largest singular value by construction. Assuming that the well-separatedness condition holds and that the cutsizes are sufficiently small, Lemma 5.1 suggests that the bi-clustering information is built into the algebraic structure in equations (25) and (26) of the singularvectors pertaining to the largest k singularvalues of $\widehat{W}^b$. See the end of Section 6 for more discussion.

6. Experiments. Document clustering has been popular in analyzing text information. We have performed experiments on newsgroup articles in the following 20 newsgroups.

Note that some of these newsgroups are completely unrelated, such as **NG1** and **NG2**; others are quite well related, such as **NG18** and **NG19**. The newsgroup dataset together with the `bow` toolkit for processing is downloaded from *http://www.cs.cmu.edu/afs/cs/project/theo-11/www/naive-bayes.html*. Similar text clustering experiments are conducted in [22].

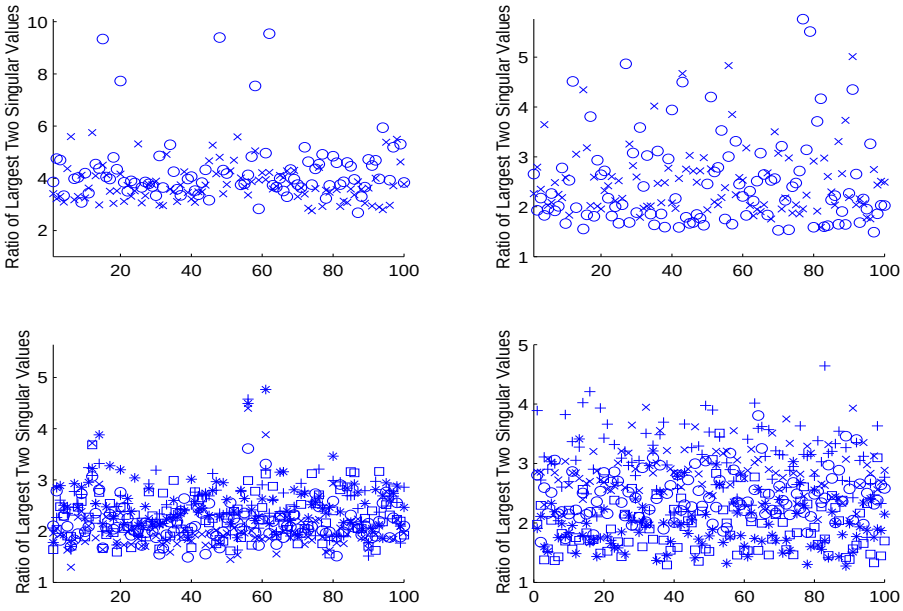
We generated a text corpus which contains 1000 articles from each newsgroup. A total of 77952 words are selected, resulting in a sparse term-document matrix with density $\approx 0.1\%$. Words are stemmed using [16]. Standard `tf.idf` scheme for term weighting is used. This corresponds to the vector-space model in information retrieval [20].

We have performed a large number of experiments on our text corpus to experimentally verify the spectral structure suggested in Lemma 4.1. Each run of our experiments consists of the following steps:

1. Take 100 documents each randomly from k newsgroups in the text corpus, and generate the term-document matrix by throwing away terms that occur in few documents and documents that have few words. The document weighting matrix is taken to be the cosine similarity[1] among documents,
2. Compute k largest eignvalues and the corresponding unit eigenvectors of the normalized weight matrix. Let $\widetilde{V}$ be the matrix of these computed eigenvectors.
3. Normalize rows of $\widetilde{V}$ so that each row has unit norm.

[1]The cosine similarity measures the distance between documents as the cosine of the angle between their corresponding vectors.

TABLE 2

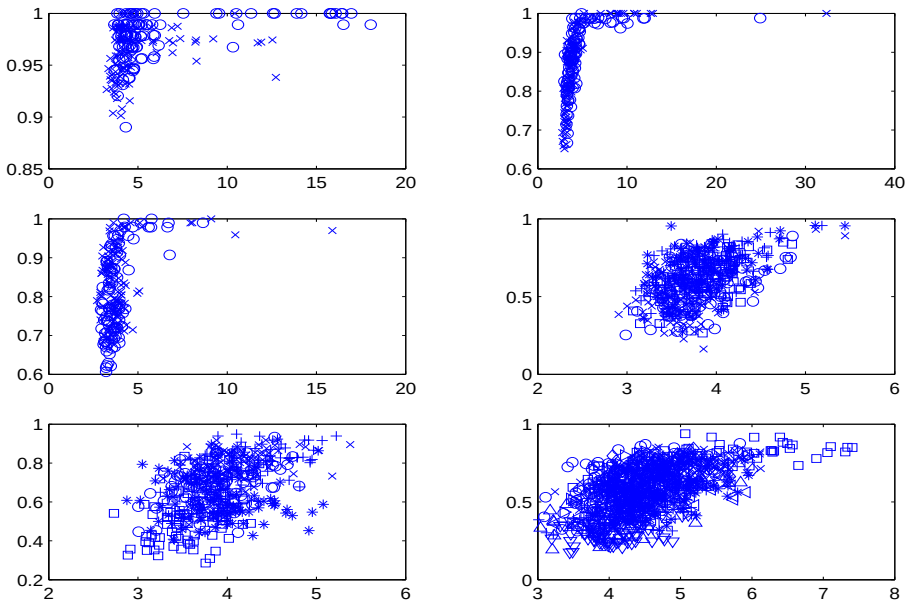


4. Partition the rows of $\widetilde{V}$ into k submatrices according to which newsgroup the corresponding document belongs.

Lemma 4.1 suggests that each of the k submatrices generated above must be nearly a rank-one matrix. In other words, the ratio of the largest two singular values for each submatrix should be very large.

In the figure above, we plot our results with various newsgroups. Different symbols (+ or o) represent different newsgroups. Each symbol represents the ratio of the 2 largest singular values for the submatrix pertaining to a particular cluster. The top left plot shows our results for 100 runs on $k = 2$ newsgroups **NG1** and **NG2**, whose contents should be largely unrelated. The plot indicates for each newsgroup and for each run, the singular value ratio is indeed much larger than 1, in good agreement with Lemma 4.1. The top right plot shows our results for 100 runs on newsgroups **NG18** vs. **N1G9**, whose contents should be somewhat related. The bottom left and bottom right plots are on two groups of 5 newsgroups each, namely **NG2**, **NG3**, **NG4**, **NG5** and **NG6** in the bottom left plot and **NG14**, **NG15**, **NG17**, **NG18** and **NG19** in the bottom right plot. In all these three plots, there are many runs where the singular value ratios are very close to 1, indicating that the structure suggested by Lemma 4.1 does not exactly exist. Hence Lemma 4.1 in general is not directly applicable to our data clustering problem, when the weighting matrix is generated in the way described in this paper.

However, it turns out that a slightly different structure does seem to exist. In our additional experiments, we tried to find a large sub-block within each submatrix for which the structure of Lemma 4.1 does exist. To this end, we choose a pre-set angular parameter $\theta = \cos^{-1}(0.85) \approx 32^o$. For each row v^T in the submatrix, we put into a sub-block all the rows that span an angle at most θ with v and compute the ratio of the largest two singular values within this sub-block. This generates a lot of



sub-blocks for each submatrix and we choose the best sub-block to be the one with the most number of rows. If Lemma 4.1 was indeed true, the best sub-block would be the same as the submatrix and the ratio of the singular values would be very large. For the text corpus that we generated, Lemma 4.1 is invalid 50% of the time. On the other hand, we have been able to find these best sub-blocks for for which the rank-one structure suggested by Lemma 4.1 does exist. In other words, the ratio of the two largest singular values for such sub-blocks is much larger than 1.

In the figure above, we plot our results with various newsgroups. Again, different symbols represent different newsgroups. The top left plot shows our results for 100 runs on $k = 2$ newsgroups **NG1** and **NG2**, the same document data used for generating the corresponding plot in the previous figure. The plot indicates for each newsgroup and for each run, the best sub-block represents about 90% or more of the original submatrix and the singular value ratio is indeed much larger than 1. The top right plot shows our results for 100 runs on newsgroups **NG8** vs. **NG9**, whose contents should be somewhat related. The plot indicates for each newsgroup and for each run, the best sub-block represents about $65 - 70\%$ or more of the original submatrix and the singular value ratio is also much larger than 1. In the middle left plot, we plot our results for 100 runs on newsgroups **NG18** and **NG19**. Here we also used the same document data that was used to generating the corresponding plot in the previous figure. The difference now is that for each newsgroup and for each run, the best sub-block represents about 65% or more of the original submatrix with a large singular value ratio. In the middle right plot, we plot our results for 100 runs on 5 newsgroups **NG2**, **NG3**, **NG4**, **NG5** and **NG6**; and in the bottom left plot we plot 100 runs on 5 newsgroups **NG14**, **NG15**, **NG17**, **NG18** and **NG19**. These are also the text data used in the previous figure. For most newsgroups and most runs, the best sub-blocks contain substantial fractions of the original submatrices with large singular value ratios. Finally, in the bottom right plot, we plot our results for 100 runs on a set of 10 newsgroups, which includes **NG1**, **NG5**, **NG7**, **NG8**, **NG11**,

NG12, **NG13**, **NG14**, **NG15** and **NG17**. Again, for most newsgroups and most runs, the best sub-blocks contain substantial fractions of the original submatrices with large singular value ratios.

Now we make a comment on Lemma 5.1. Since it is not known a priori what the keywords should be for each data cluster, explicitly verifying the algebraic structure in Lemma 5.1 is problematic for our newsgroup data. However, our experiments do confirm that sub-blocks with rank-one structures do indeed exist within largest k left and right singular vectors.

7. Conclusions and Future Work. In this paper, we have shown that important algebraic structures in general exist in the eigenvector and singular vector matrices for data clusters for both graph clustering and bi-clustering. Future work includes development of efficient and effective spectral methods that fully take advantage of such structures for cluster analysis and data mining.

REFERENCES

[1] C. J. Alpert, A. B. Kahng and D. S. Yao. Spectral Partitioning With Multiple Eigenvectors, *Discrete Applied Mathematics,* 90:3-26, 1999.

[2] Y. Azar, A. Fiat, A. Karlin, F. McSherry and J. Saia, Spectral Analysis of Data, *Symposium on Theory of Computation*, 2001.

[3] P. K. Chan, M. D. F. Schlag and J. Y. Zien. Spectral based multi-way FPGA partitioning, In *Int'l Symposium on Fied-Programmable Gate Arrays*, pp. 133-139, ACM, 1995.

[4] F. R. K. Chung. *Spectral Graph Theory.* Amer. Math. Society., 1997.

[5] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. *Univ Texas Austin CS Tech Report TR 2001-05*, 2001.

[6] C. Ding, X. He and H. Zha. A Spectral Method to Separate Disconnected and Nearly-disconnected Web Graph Components. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 275-280, 2001.

[7] C. Ding, X. He, H. Zha, M. Gu and H. Simon. Spectral Min-max Cut for Graph Partitioning and Data Clustering. To appear in *Proceedings of the First IEEE International Conference on Data Mining*, 2001.

[8] P. Drineas, A. Frieze, R. Kannan, S. Vempala and V. Vinay. Clustering in large matrices and Graphs. *Proc. of the 10th ACM-SIAM Symposium on Discrete Algorithms*, 1999.

[9] M. Fiedler. Algebraic connectivity of graphs. *Czech. Math. J.*, 23:298–305, 1973.

[10] L. Hagen and A.B. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE. Trans. on Computed Aided Desgin*, 11:1074–1085, 1992.

[11] X. He, H. Zha, C. Ding, and H.D. Simon. Web document clustering using hyperlink structures. *Tech Report CSE-01-006, Department of Computer Science and Engineering, Pennsylvania State University, to appear in special issue of Computational Statistics and Data Analysis*, April 2001.

[12] B. Hendrickson and R. Leland. Chaco mesh partitioning software.

[13] R. Kannan, S. Vempala and A. Vetta. On clusterings: good, bad and spectral. *Proc. of the 41st Foundations of Computer Science*, 2000.

[14] G. Karypis and V. Kumar. Metis graph partitioning software.

[15] G. Karypis and V. Kumar. Parallel multilevel k-way partitioning scheme for irregular graphs. *Siam Review*, 41:278–300, 1999.

[16] A. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. *http://www.cs.cmu.edu/mccallum/bow*, 1996.

[17] M. I. Ng, A. Y. Jordan and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, 14, 2001.

[18] A. Pothen, H. D. Simon, and K. P. Liou. Partitioning sparse matrices with egenvectors of graph. *SIAM Journal of Matrix Anal. Appl.*, 11:430–452, 1990.

[19] C. Papadimitriou, P. Raghvan, S. Rajagopalan and A. Tomkins, Latent Semantic Indexing: A Probabilistic Analysis. *Proceedings of ACM Symposium on Principles of Database Systems*, 1997.

[20] G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[21] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE. Trans. on Pattern Analysis and Machine Intelligence*, 2000.

[22] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. *Proceedings of SIGIR-2000*, 2000.

[23] G. W. Stewart, Matrix Algorithms, Volume 1: Basic Decompositions, *SIAM*, 1998.

[24] H. Zha, X. He, C. Ding, M. Gu and H. Simon. Bipartite Graph Partitioning and Data Clustering. To appear in Proceedings of ACM CIKM 2001, the Tenth International Conference on Information and Knowledge Management, 2001.

[25] H. Zha and Z. Zhang, A note on constructing a symmetric matrix with specified diagonal entries and eigenvalues, *BIT*, Vol 35(446-452), 1995.