SANDIA REPORT

SAND2006-2079
Unclassified Unlimited Release
Printed April 2006

Multilinear algebra for analyzing data with multiple linkages

Daniel M. Dunlavy, Tamara G. Kolda, and W. Philip Kegelmeyer

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.doe.gov/bridge

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/ordering.htm

SAND2006-2079
Unclassified Unlimited Release
Printed April 2006

Multilinear algebra for analyzing data with multiple linkages

Daniel M. Dunlavy
Optimization & Uncertainty Estimation Department
Sandia National Laboratories
Albuquerque, NM 87185–1110
dmdunlavy@sandia.gov

Tamara G. Kolda, W. Philip Kegelmeyer
Computational Science and Mathematics Science Research Department
Sandia National Laboratories
Livermore, CA 94551-9159
{tgkolda,wpk}@sandia.gov

Abstract

Link analysis typically focuses on a single type of connection, e.g., two journal papers are linked because they are written by the same author. However, often we want to analyze data that has multiple linkages between objects, e.g., two papers may have the same keywords and one may cite the other. The goal of this paper is to show that multilinear algebra provides a tool for multi-link analysis. We analyze five years of publication data from journals published by the Society for Industrial and Applied Mathematics. We explore how papers can be grouped in the context of multiple link types using a tensor to represent all the links between them. A PARAFAC decomposition on the resulting tensor yields information similar to the SVD decomposition of a standard adjacency matrix. We show how the PARAFAC decomposition can be used to understand the structure of the document space and define paper-paper similarities based on multiple linkages. Examples are presented where the decomposed tensor data is used to find papers similar to a body of work (e.g., related by topic or similar to a particular author's papers), find related authors using linkages other than explicit co-authorship or citations, distinguish between papers written by different authors with the same name, and predict the journal in which a paper was published.

Contents

Figures

Tables

Multilinear algebra for analyzing data with multiple linkages

1 Introduction

Data with multiple link types is challenging to analyze, yet such data abounds. For example, Adamic and Adar [2] analyze a social network where nodes are connected by organizational structure, i.e., each employee is connected to his or her boss, and also by direct email communication. Social networks clearly have many types of links — familial, communication (phone, email, etc.), organizational, geographical, etc. Some links are explicit, e.g., siblings, while others are implicit, e.g., two people may be implicitly linked if they work at the same company.

Our focus is on journal publication data—specifically considering the many ways that two papers may be linked. We analyze five years of journal publication data from eleven journals and a set of conference proceedings published by the Society for Industrial and Applied Mathematics (SIAM). Explicit, directed links exist whenever one paper cites another. Implicit links are derived based on title, abstract, and keyword similarities as well as author similarity. Historically, bibliometric researchers have focused solely on citation analysis or text analysis, but not both simultaneously.

Our overarching goal is to analyze data with multiple explicit and implicit links, and to derive feature vectors. Though we focus on bibliometric analysis, the techniques that we discuss are potentially applicable to a wide range of tasks and have already been used for higher-order web link analysis [23, 22].

Many link analysis techniques, e.g., PageRank [8] and HITS [20], are focused on a single link type and decompose the adjacency matrix of the graph. Our method is similar, except that we allow multiple link types and decompose the adjacency *tensor* of the graph. We define a three-way array of size $N \times N \times K$ where N is the number of nodes (e.g., documents) and K is the number of different link types. Thus, the (i, j, k) entry is nonzero if node i is connected to node j by link type k. In the example of Adamic and Adar [2] discussed above, we have two links types where $k = 1$ reflects the organization connection and $k = 2$ reflects the email communication connection. For our data, we consider five different link types based on abstract, title, and keyword similarity as well as common authors and citations; see Figure 1.

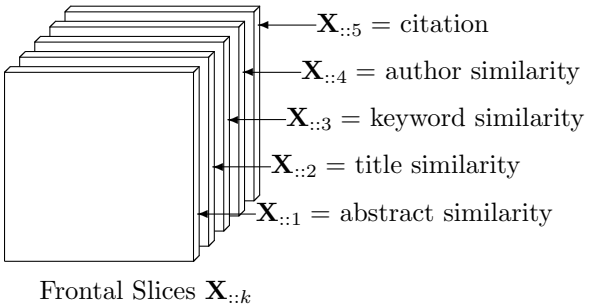

Figure 1. Tensor slices capture different link types

We decompose the tensor using PARAFAC [17, 9] and use the results to understand the data in several ways.

• The raw factors reveal "communities" within the article collection and how they are connected, e.g., the connection may be mostly due to title similarity in some cases and citation similarity in other cases.

• The component matrices of the factorization provide derived feature vectors which are, in turn, used to derive a latent similarity measure for the documents. We demonstrate that this can be used to compute similarity scores for documents that combine the multiple linkage types.

• More generally, the most similar papers to a *body of work*, e.g., by a given author, are analyzed to find the most similar papers in the larger collection.

• We also show that the feature vectors associated with individual authors can be used to disambiguate authors. For example, is H. Simon the same as H. S. Simon?

• We apply supervised learning techniques (decision trees and ensembles) to the derived feature vectors to predict which journals published which papers.

2 Related Work

2.1 Analysis of publication data

Researchers look at publication data to understand the impact of individual authors and who is collaborating with whom, to understand the type of information being published and by which venues, and to extract "hot topics" and understand trends [7]. For example, Barábasi et al. [4] consider the social network of scientific collaborations based on publication data, particularly the properties of the entire network and its evolution over time.

The 2003 KDD Cup challenge was to apply data mining techniques to bibliographic data [14]. McGovern et al. [25] looked at a number of questions. Of particular relevance to this paper is that, in their community analysis, they found that clustering based only on text did not yield useful clusters. Instead, they used spectral-based citation analysis but weighted the links on the citation graph by the cosine similarity of the paper abstracts. This is a novel method for incorporating text similarity information into the citation graph. Additionally, for predicting where an article will be published, they use relational probability trees (see, e.g., [16]). Lin and Chalupsky [10] consider the idea of finding interesting connections within a graph that represents papers, authors, organizations, etc., without restricting the connection to be of a certain type. The goal of their "novel node discovery" and our methods is quite similar—to find related nodes—but the techniques are very different. They analyze the graph directly whereas we analyze a latent representation. Hill and Provost [19] use *only* citation information to predict authorship almost half the time. They note that including text-based methods may improve accuracy.

2.2 Higher-order analysis

Tensor decompositions have a long history and have been used in applications ranging from chemometrics [27] to image analysis [32]. Recently, they have been applied to data-centric problems including analysis of click-through data, using an alternate decomposition known as Tucker [28], and chatroom analysis comparing different tensor decompositions [1]. Tao et al. [29] do supervised learning on a low-rank PARAFAC-like decomposition of their data. Liu et al. [24] consider a Tucker decomposition for text classification. Kolda et al. [23, 22] used the PARAFAC decomposition to extend the well-known HITS method to incorporate anchor text information.

2.3 Other related work

Relational probability trees (RPTs) [16, 15] offer another technique for analyzing graphs with different link and node types. These methods can be used as alternatives, particularly in prediction tasks such as determining the journal that an article will be published in. Rattigan and Jensen [26] have also used RPTs for finding anomalous links.

As we later consider the problem of disambiguation, we note that Bekkerman and McCallum [5] approach this issue by considering the appearance of individuals on the web.

3 The PARAFAC decomposition

3.1 Notation

Scalars are denoted by lowercase letters, e.g., c. Vectors are denoted by boldface lowercase letters, e.g., $\mathbf{v}$. The ith entry of $\mathbf{v}$ is denoted by v_i. Matrices are denoted by boldface capital letters, e.g., $\mathbf{A}$. The jth column of $\mathbf{A}$ is denoted by $\mathbf{a}_j$ and element (i, j) by a_{ij}. Tensors (i.e., multi-way arrays) are denoted by boldface Euler script letters, e.g., $\mathcal{X}$. Element (i, j, k) of a 3rd-order tensor $\mathcal{X}$ is denoted by x_{ijk}. The symbol $\circ$ denotes the outer product of vectors; for example, if $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^J$, $\mathbf{c} \in \mathbb{R}^K$, then $\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$ if and only if $x_{ijk} = a_i b_j c_k$ for all $1 \leq i \leq I$, $1 \leq j \leq J$, $1 \leq k \leq K$. The symbol $\otimes$ denotes the Kronecker product of vectors; for example, $\mathbf{x} = \mathbf{a} \otimes \mathbf{b}$ means $x_\ell = a_i b_j$ with $\ell = j + (i-1)(J)$ for all $1 \leq i \leq I$, $1 \leq j \leq J$. The symbol $*$ denotes the Hadamard (i.e., elementwise) matrix product.

The norm of a tensor is given by the square root of the sum of the squares of all its elements, i.e., for a tensor $\mathcal{X}$ of size $I \times J \times K$, $\| \mathcal{X} \|^2 \equiv \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} x_{ijk}^2$. This is the higher-order analogue of the matrix Frobenius norm.

3.2 PARAFAC Description

The PARAFAC decomposition [17, 9] is a higher-order analogue of the matrix singular value decomposition (SVD) which decomposes a tensor into a sum of rank-1 tensors. PARAFAC should not be confused with the Tucker decomposition [31], a different higher-order analogue of the SVD.

Recall that our data will be arranged into a tensor $\mathcal{X}$ of size $N \times N \times K$. We choose a desired rank of the approximation, R, which loosely reflects the number of *communities* in the data. Often some experimentation is required to determine the most useful value of R, as will be seen in §5. The goal is to approximate $\mathcal{X}$ as

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{h}_r \circ \mathbf{a}_r \circ \mathbf{c}_r \equiv \boldsymbol{\lambda} \, [\![\mathbf{H}, \mathbf{A}, \mathbf{C}]\!].$$

Here, $\boldsymbol{\lambda}$ is a vector of length R that represents the weight of each community; $\mathbf{H}$ and $\mathbf{A}$ are matrices of size $N \times R$ that represent the hub and authority score for each document with respect to each community, and the matrix $\mathbf{C}$ of size $K \times R$ represents the importance of each link type with respect to each community. The matrices $\mathbf{H}$, $\mathbf{A}$, $\mathbf{C}$ have columns of length one; but, in contrast to the solution provided by the SVD, these columns are not generally orthonormal [21]. The interpretation of these quantities will be made clearer in §5.1. The PARAFAC decomposition approximates the tensor $\mathcal{X}$ by the sum of R rank-1 outer products, as shown in Figure 2.

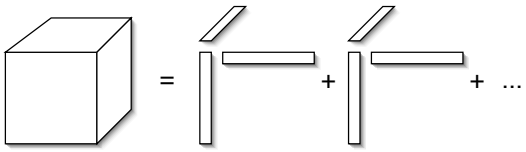

Figure 2. The PARAFAC decomposition separates the tensor into rank-1 factors.

3.3 PARAFAC Algorithm

A common approach to solving the PARAFAC model is the use of alternating least squares (ALS) [17, 13, 30]. At each inner iteration, we solve for one component matrix while holding the others fixed. For example, suppose we wish to solve for the matrix $\mathbf{C}$ when $\mathbf{H}$ and $\mathbf{A}$ are fixed, i.e.,

$$\min_{\mathbf{C}} \| \mathcal{X} - [\![\mathbf{H}, \mathbf{A}, \mathbf{C}]\!] \| . \tag{1}$$

In this case, we omit $\boldsymbol{\lambda}$ because it will just be absorbed into the lengths of the columns of $\mathbf{C}$ when the computation is complete. Equation (1) can be rewritten as a matrix problem (see, e.g., [27])

$$\min_{\mathbf{C}} \left\| \mathbf{X}_{(3)} - \mathbf{C} \left(\mathbf{A} \odot \mathbf{H} \right) \right\| . \tag{2}$$

Here $\mathbf{X}_{(3)}$ is the mode-3 matritization or unfolding (see, e.g., []) of the tensor $\mathcal{X}$, i.e.,

$$\left(\mathbf{X}_{(3)} \right)_{kl} = x_{ijk} \text{ where } l = (i-1)(N-1) + j + 1.$$

The notation $\mathbf{A} \odot \mathbf{H}$ is the Khatri-Rao product [27] and denotes the columnwise Kronecker product, i.e.,

$$\mathbf{A} \odot \mathbf{H} = \begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{h}_1 & \mathbf{a}_2 \otimes \mathbf{h}_2 & \cdots & \mathbf{a}_R \otimes \mathbf{h}_R \end{bmatrix} .$$

The pseudo-inverse of a Khatri-Rao product is given by

$$(\mathbf{A} \odot \mathbf{H})^{\dagger} = (\mathbf{A} \odot \mathbf{H})^{\mathsf{T}} (\mathbf{A}^{\mathsf{T}} \mathbf{A} * \mathbf{H}^{\mathsf{T}} \mathbf{H})^{\dagger},$$

so that only the pseudo-inverse on an $R \times R$ matrix needs to be calculated rather than that of an $N^2 \times R$ matrix [27]. The optimal $\mathbf{C}$ is the least squares solution:

$$\mathbf{C} = \mathbf{X}_{(3)} \left(\mathbf{A} \odot \mathbf{H} \right)^{\dagger} ,$$

which can be computed efficiently thanks to the properties of the Khatri-Rao product. The other component matrices can be computed in an analogous fashion.

It is generally efficient to initialize the ALS algorithm with the leading R leading eigenvalues of $\mathbf{X}_{(n)} \mathbf{X}_{(n)}^{\mathsf{T}}$ for the nth component matrix; see, e.g., [22]. In this case, however, $\mathbf{X}_{(3)} \mathbf{X}_{(3)}^{\mathsf{T}}$ is only of size $K \times K$ and K is generally much smaller than R, so we cannot use this scheme to initialize $\mathbf{C}$. Consequently, we can only use this scheme to initialize $\mathbf{A}$ and $\mathbf{H}$. Therefore, we begin by solving for $\mathbf{C}$ using the initialized values of $\mathbf{A}$ and $\mathbf{H}$. The resulting algorithm is presented in Algorithm 1.

In the discussion that follows, we let $\boldsymbol{\Lambda}$ denote the $R \times R$ diagonal matrix whose diagonal is $\boldsymbol{\lambda}$.

4 The Data

The data consists of ISI publication metadata from eleven SIAM journals as well as SIAM proceedings (SIAM PROC S) for the period 1999–2004. There are 5022 articles; the number of articles per

Algorithm 1 Alternating Least Squares (ALS)

in: Tensor $\mathcal{X}$ of size $N \times N \times K$
in: Desired rank $R > 0$.
$\mathbf{H} \leftarrow R$ principal eigenvalues of $\mathbf{X}_{(1)}\mathbf{X}_{(1)}^{\mathsf{T}}$
$\mathbf{A} \leftarrow R$ principal eigenvalues of $\mathbf{X}_{(2)}\mathbf{X}_{(2)}^{\mathsf{T}}$
Initialize $\mathbf{A}$ to be ...
repeat
  $\mathbf{C} \leftarrow \mathbf{X}_{(3)}(\mathbf{A} \odot \mathbf{H})^{\mathsf{T}} \left(\mathbf{A}^{\mathsf{T}}\mathbf{A} * \mathbf{H}^{\mathsf{T}}\mathbf{H}\right)^{\dagger}$
  Normalize columns of $\mathbf{C}$ to length 1
  $\mathbf{A} \leftarrow \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{H})^{\mathsf{T}} \left(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{H}^{\mathsf{T}}\mathbf{H}\right)^{\dagger}$
  Normalize columns of $\mathbf{A}$ to length 1
  $\mathbf{H} \leftarrow \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{A})^{\mathsf{T}} \left(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A}\right)^{\dagger}$
  Store column norms of $\mathbf{H}$ in $\boldsymbol{\lambda}$ and normalize columns of $\mathbf{H}$ to length 1
until the fit ceases to improve or the maximum number of iterations is exceeded.
out: $\boldsymbol{\lambda} \in \mathbb{R}^R$, $\mathbf{A}, \mathbf{H} \in \mathbb{R}^{N \times R}$ and $\mathbf{C} \in \mathbb{R}^{K \times R}$ such that $\mathcal{X} \approx \boldsymbol{\lambda}\,[\![\mathbf{H}, \mathbf{A}, \mathbf{C}]\!]$.

publication is shown in Table 14. The names of the journals used throughout this paper are the ISI abbreviations[1] for the journals.

4.1 The Data as a Tensor

The data is represented as a $N \times N \times K$ tensor where N is the number of documents and K is the number of link types. In our case, $N = 5022$ and $K = 5$. The five link types are as follows; see also Figure 1.

(1) The first slice ($\mathbf{X}_{::1}$) represents abstract similarity, i.e., x_{ij1} is the cosine similarity of the abstracts for documents i and j. The similarities were computed as follows. We used the Text to Matrix Generator (TMG) [33] to generate a term-document matrix, $\mathbf{T}$. Note that we removed all words starting with a number and those words appearing on the default TMG stopword list. We used term frequency and inverse document frequency local and global weightings (tf.idf); this means that

$$t_{ij} = f_{ij} \log_2(N/N_i)$$

where f_{ij} is the frequency of term i in document j and N_i is the number of documents that term i appears in. Each column of $\mathbf{T}$ is normalized to length one (for cosine scores), and then we set

$$\mathbf{X}_{::1} = \mathbf{T}^{\mathsf{T}}\mathbf{T}.$$

Because they are cosine scores, all are in the range [0,1]. In order to sparsify the slice, only scores greater than 0.2 (chosen heuristically to reduce the total number of nonzeros in all three text similarity slices to approximately 250,000) are retained.

(2) The second slice ($\mathbf{X}_{::2}$) represents title similarity, i.e., x_{ij2} is the cosine similarity of the titles for documents i and j. It is computed in the same manner as the abstract similarity slice.

(3) The third slice ($\mathbf{X}_{::3}$) represents author-supplied keyword similarity, i.e., x_{ij3} is the cosine similarity of the keywords for documents i and j. It is computed in the same manner as the abstract similarity slice.

(4) The fourth slice ($\mathbf{X}_{::4}$) represents author similarity, i.e., x_{ij4} is the similarity of the authors

[1] http://www.isiknowledge.com/

for documents i and j. It is computed as follows. Let $\mathbf{W}$ be the author-document matrix such that

$$w_{ij} = \begin{cases} 1/\sqrt{M_j} & \text{if author } i \text{ wrote document } j \\ 0 & \text{otherwise.} \end{cases}$$

Here M_j is the number of authors for document j. Thus,

$$\mathbf{X}_{::4} = \mathbf{W}^\mathsf{T}\mathbf{W}.$$

(5) The fifth slice ($\mathbf{X}_{::5}$) represents citation information, i.e.,

$$x_{ij5} = \begin{cases} 2 & \text{if document } i \text{ cites document } j \\ 0 & \text{otherwise.} \end{cases}$$

For this document collection, a weight of 2 was chosen heuristically so that the overall *slice weight* (i.e., the sum of all the entries in $\mathbf{X}_{::k}$, see Table 2) would not be too small relative to the other slices. The interpretation is that there are relatively few connections in this slice, but each indicates a strong connection. In future work, we would like to consider less ad hoc ways of determining the value for citation links.

These particular choices for link type and particular similarity measures are an example of what can be done — many other choices are possible. For instance, non-symmetric similarity weights are an option; e.g., if document A is a subset of document B, we could say that B is very similar to A but A is not so similar to B because B talks about other topics as well. Moreover, we could use a symmetric citation measure such as co-citation. We could include other connections such as whether or not two articles are published in the same journal. We may also wish to consider data about when an article is published, and so on.

4.2 Quantitative measurements on the data

Table 1 shows overall statistics on the dataset. The columns labeled *Total* and *Max* show the totals over the dataset and the maximum values per article, respectively. Note that we only consider citations within the dataset. Table 14 shows the total number of articles per journal. The most citations *to* a single article is 15.

Table 1. Data statistics

Number of publications	12	
Number of documents	5022	
	Total	*Max*
Unique terms	16617	831
abstracts	15752	802
titles	5164	33
keywords	5248	40
Authors	6891	13
Citations	2659	12

Table 2 shows the number of nonzero entries and the sums of the entries for each slice. The text similarity slices ($k = 1, 2, 3$) have large numbers of nonzeros but low average values, the author similarity slice has few nonzeros but a higher average value, and the citation slice has the fewest nonzeros but all values are 2.

Table 2. Per slice data

Slice (k)	*Description*	*Nonzeros*	$\sum_i \sum_j x_{ijk}$
1	Abstract Similarity	28476	7695.28
2	Title Similarity	120236	33285.79
3	Keyword Similarity	115412	16201.85
4	Author Similarity	16460	8027.46
5	Citation	2659	5318.00

5 Numerical Results

We compute the PARAFAC decomposition of $\mathcal{X}$ using different ranks (choices for R). The resulting component matrices provide feature vectors in R-dimensional space for each document, which can be used as described in the sections that follow.

5.1 Community Identification

The PARAFAC factors themselves reveal interesting connections in the data. The largest entries for the vectors in each factor,

$$\{\mathbf{h}_r, \mathbf{a}_r, \mathbf{c}_r\}$$

can be interpreted as a sort of interlinked community. For the rth factor, high-scoring hubs point to high-scoring authorities with the high-scoring link types in that factor.

Table 3. Link scores for two of the PARAFAC factors for $R = 30$.

$r = 1$	
Score	*Link Type*
0.95	TitleSim
0.28	KeywordSim
0.07	AbstractSim
0.06	Citation
0.06	AuthorSim

$r = 10$	
Score	*Link Type*
0.96	Citation
0.19	AuthorSim
0.16	TitleSim
0.10	KeywordSim
0.06	AbstractSim

For example, the first set ($r = 1$) of link scores is shown in Table 3, which is a sorted version of $\mathbf{c}_1$, the first column of $\mathbf{C}$. Title similarity has the highest score, as well as a strong keyword similarity. In fact, the top three link types are based on text similarity and so are symmetric. Therefore, it is no surprise that the hubs and authorities (the largest entries of $\mathbf{h}_1$ and $\mathbf{a}_1$ shown in Table 4) are the same papers, just ordered slightly differently. This cluster of papers is clearly about conservation laws.

On the other hand, the tenth factor ($r = 10$) has citation as the dominant link type; see Table 3. Citation links are not symmetric, so the hubs and authorities are less similar; see Table 5. This factor appears to be about preconditioning, though the titles do not all share keywords. Moreover, the third authority is not directly about preconditioning but rather about graph partitioning, which is often used in preconditioning, and thus highly cited.

The choice to have symmetric or non-symmetric connections affects the output and interpretation of the PARAFAC model. In this case, we have mixed mostly symmetric connections with one

Table 4. Highest scoring hubs & authorities for first PARAFAC factor for $R = 30$.

Hubs	
Score	*Title*
0.18	On the boundary control of systems of conservation laws
0.17	On stability of conservation laws
0.16	Two a posteriori error estimates for one-dimensional scalar conservation laws
0.16	A free boundary problem for scalar conservation laws
0.15	Convergence of SPH method for scalar nonlinear conservation laws
0.15	Adaptive discontinuous Galerkin finite element methods for nonlinear hyperbolic conservation laws
0.15	High-order central schemes for hyperbolic systems of conservation laws
0.15	Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws
Authorities	
Score	*Title*
0.18	On the boundary control of systems of conservation laws
0.18	On stability of conservation laws
0.16	Two a posteriori error estimates for one-dimensional scalar conservation laws
0.16	A free boundary problem for scalar conservation laws
0.16	Adaptive discontinuous Galerkin finite element methods for nonlinear hyperbolic conservation laws
0.16	Convergence of SPH method for scalar nonlinear conservation laws
0.15	Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws
0.14	High-order central schemes for hyperbolic systems of conservation laws

asymmetric connection. In cases where all or most of the factors are symmetric, one option is to constrain the first two factors to be identical, and this is a topic for future research.

The benefit of the multi-link analysis is that each different factor potentially has a different set of important link types, which helps in understanding more complex connections than might be identified with a single analysis, e.g., just citation analysis or just text similarity.

5.2 Latent Document Similarity

The authority and hub matrices provide latent representations of each document in terms of the principal factors in the PARAFAC analysis. This, in turn, yields a latent document similarity score that provides information beyond direct text, author, or citation analysis. The $N \times N$ similarity matrix is computed as

$$\mathbf{S} = \frac{1}{2}\mathbf{H}\mathbf{H}^{\mathsf{T}} + \frac{1}{2}\mathbf{A}\mathbf{A}^{\mathsf{T}}.$$

The similarity for documents i and j is given by s_{ij}. We could alternatively use just the hub matrix $\mathbf{H}$ or just the authority matrix $\mathbf{A}$ or some other combination of the two matrices. Emphasizing the $\mathbf{H}$ matrix chooses papers that cite the same papers, whereas emphasizing $\mathbf{A}$ chooses papers that are cited by the same papers. Moreover, we could incorporate $\mathbf{\Lambda}$, e.g.,

$$\mathbf{S} = \frac{1}{2}\mathbf{H}\mathbf{\Lambda}\mathbf{H}^{\mathsf{T}} + \frac{1}{2}\mathbf{A}\mathbf{\Lambda}\mathbf{A}^{\mathsf{T}}.$$

For now, we relegate these issues to future study.

These sorts of issues are reminiscent of the choices facing users of latent semantic indexing (LSI) [12] which uses the SVD of a term-document matrix, producing term and document matrices. There is a choice of how to use the diagonal scaling for the queries and comparisons [6].

Table 5. Highest hubs & authorities for the tenth PARAFAC factor for $R = 30$.

Hubs	
Score	*Title*
0.36	Multiresolution approximate inverse preconditioners
0.20	Preconditioning highly indefinite and nonsymmetric matrices
0.16	A factored approximate inverse preconditioner with pivoting
0.16	On two variants of an algebraic wavelet preconditioner
0.14	A robust and efficient ILU that incorporates the growth of the inverse triangular factors
0.11	An algebraic multilevel multigraph algorithm
0.11	On algorithms for permuting large entries to the diagonal of a sparse matrix
0.11	Preconditioning sparse nonsymmetric linear systems with the Sherman-Morrison formula
Authorities	
Score	*Title*
0.27	Ordering anisotropy and factored sparse approximate inverses
0.25	Robust approximate inverse preconditioning for the conjugate gradient method
0.23	A fast and high-quality multilevel scheme for partitioning irregular graphs
0.20	Orderings for factorized sparse approximate inverse preconditioners
0.19	The design and use of algorithms for permuting large entries to the diagonal of sparse matrices
0.17	BILUM Block versions of multielimination and multilevel ILU preconditioner for general sparse linear systems
0.16	Orderings for incomplete factorization preconditioning of nonsymmetric problems
0.15	Preconditioning highly indefinite and nonsymmetric matrices

We consider the example of computing the similarity to the paper *Link analysis: Hubs and authorities on the World Wide Web*. The results depend on the choice of R, i.e., the number of factors used in the PARAFAC decomposition. Table 6 shows the result for $R = 10$ and Table 7 for $R = 30$. The $R = 10$ case is not very precise, citing a variety of papers as related, ranging from the topic of sparse approximate inverses (arguably distantly related) to interior point methods (not related) and graph partitioning (related). Moreover, note that the similarity scores are very low. The results for $R = 30$ appears to be much improved because the focus is on papers about graphs; unfortunately, this dataset does not contain many papers about web analysis.

Just as in LSI, choosing the rank of the approximation (R) is heuristic.

Table 6. Article similarities to *Link Analysis: Hubs and Authorities on the World Wide Web* using a rank $R = 10$ PARAFAC decomposition.

Score	*Title*
0.000079	Ordering anisotropy and factored sparse approximate inverses
0.000079	Robust approximate inverse preconditioning for the conjugate gradient method
0.000077	An interior point algorithm for large-scale nonlinear programming
0.000073	Primal-dual interior-point methods for semidefinite programming in finite precision
0.000068	Some new search directions for primal-dual interior point methods in semidefinite programming
0.000068	A fast and high-quality multilevel scheme for partitioning irregular graphs
0.000067	Reoptimization with the primal-dual interior point method
0.000065	Superlinear convergence of primal-dual interior point algorithms for nonlinear programming
0.000064	A robust primal-dual interior-point algorithm for nonlinear programs
0.000063	Orderings for factorized sparse approximate inverse preconditioners

Table 7. Article similarities to *Link Analysis: Hubs and Authorities on the World Wide Web* using a rank $R = 30$ PARAFAC decomposition.

Score	*Title*
0.000563	Skip graphs
0.000356	Random lifts of graphs
0.000354	A fast and high-quality multilevel scheme for partitioning irregular graphs
0.000322	The minimum all-ones problem for trees
0.000306	Rankings of directed graphs
0.000295	Squarish k-d trees
0.000284	Finding the k-shortest paths
0.000276	On floor-plan of plane graphs
0.000275	1-Hyperbolic graphs
0.000269	Median graphs and triangle-free graphs

5.3 Analyzing a Body of Work via Centroids

Finding documents similar to a body of work may be useful in a literature search or in finding other authors working in a given area. We performed two sets of experiments using centroids to analyze a body of work.

In the first set of experiments, we focused on finding collections of articles containing a particular term (or phrase). We found all articles containing the term in either the title, abstract, or keywords. We computed the centroid $\mathbf{c}_h$ using the columns of the hub matrix $\mathbf{H}$ for the identified articles, and the centroid $\mathbf{c}_a$ using the columns of the authority matrix $\mathbf{A}$ for the identified articles. The similarity scores for all documents to the body of work are then computed as

$$\mathbf{s} = \frac{1}{2}\mathbf{H}\mathbf{c}_h + \frac{1}{2}\mathbf{A}\mathbf{c}_a.$$

Consequently, s_i is the similarity of the ith document to the centroid.

Table 8 shows the results of a search on the term "GMRES", which is a method of solving linear systems. In the table, we list the top scoring documents using the hub and authority matrices separately to illustrate the potential advantage of the combined score above. We do not want to overemphasize the papers that cite many of the papers about GMRES (as in the hubs here) or those which are most cited (as in the authorities here). Rather, a combination of the two, which takes into account the content of the papers (i.e., abstracts, titles, and keywords) to a greater extent than either of these two extremes. Thus, the average scores as computed above result in a more balanced look at papers about GMRES (each article denoted with an asterisk in Table 8 is one of the top ten scoring using the combined scores).

Table 9 shows the results of a search on the articles written by V. Kumar. We found all of the documents written by the author, generating the centroid and similarity score vector as above. In these 10 articles, only three papers (including the two authored by V. Kumar) are explicitly linked to V. Kumar by co-authorship or citations. Furthermore, we see several papers that are closely related to those written by V. Kumar focused on graph analysis and some that are not so obviously linked. We list the authors in Table 9 as well to illustrate that such results could be used as a starting point for finding authors related to V. Kumar that are not necessarily linked by co-authorship or citation. In this case, the author W. P. Tang appears to linked to V. Kumar.

Centroids can be a useful tool in understanding small collections of documents. As we discuss in §5.5, centroids for larger sets of documents did not prove to be as useful. Centroid-matching is potentially a useful tool in matching referees to papers. For example, program committee chairs would create a centroid for each participant on a program committee. Work assignments can be

Table 8. Articles similar to the centroid of articles containing the term *GMRES* using the hub and authority matrices to compute similarity scores.

Hubs		
Score		
$\mathbf{Hc}_h$	$\mathbf{Ac}_a$	*Title*
0.0240	0.0019	*Flexible inner-outer Krylov subspace methods
0.0185	0.0082	*FQMR A flexible quasi-minimal residual method with inexact preconditioning
0.0169	0.0017	*Theory of inexact Krylov subspace methods and applications to scientific computing
0.0132	0.0024	*GMRES with deflated restarting
0.0127	0.0003	*A case for a biorthogonal Jacobi-Davidson method Restarting and correction equation
0.0107	0.0010	A class of spectral two-level preconditioners
0.0076	0.0011	An augmented conjugate gradient method for solving consecutive symmetric positive definite linear systems
Authorities		
Score		
$\mathbf{Ac}_a$	$\mathbf{Hc}_h$	*Title*
0.0217	0.0011	*Adaptively preconditioned GMRES algorithms
0.0158	0.0014	*Inexact preconditioned conjugate gradient method with inner-outer iteration
0.0149	0.0074	*Truncation strategies for optimal Krylov subspace methods
0.0113	0.0056	*Flexible conjugate gradients
0.0082	0.0185	*FQMR A flexible quasi-minimal residual method with inexact preconditioning
0.0080	0.0007	Linear algebra methods in a mixed approximation of magnetostatic problems
0.0063	0.0060	*On the convergence of restarted Krylov subspace methods

* In top 10 scores of $\mathbf{s} = \frac{1}{2}\mathbf{Hc}_h + \frac{1}{2}\mathbf{Ac}_a$.

expedited by automatically matching articles to the appropriate experts on the committee.

As a segue to the next section, we note that finding a set of documents associated with a particular author is not always straightforward. In fact, in the example above, there is also an author named V. S. A. Kumar, and it is not clear from article titles alone that this is not the same author as V. Kumar. In the next section, we discuss the use of the feature vectors produced by tensor decompositions for author disambiguation.

5.4 Author Disambiguation

A challenging problem in working with publication data is determining whether two authors are in fact a single author using multiple aliases. Such problems are often caused by incomplete or incorrect data or varying naming conventions for authors used by different publications (e.g., James R. Smith versus J. Smith). In the SIAM articles, there are many instances where two or more authors share the same last name and at least the same first initial, e.g., V. Torczon and V. J. Torczon. In these cases, we are interested in determining which authors were listed in the data under multiple aliases.

For each group of authors sharing the same last name and same first initial, we computed centroids of the columns of the authority matrix, $\mathbf{A}$, of the rank $R = 30$ PARAFAC decomposition, corresponding to the articles written by each author. We then computed pairwise inner products of these centroids to determine which centroids were similar. (Note that the scores are values in [0,1].) The higher the similarity score, the most confident we were that two author names referred to a single person.

Table 9. Papers similar to those by V. Kumar.

Score	*Authors*	*Title*
0.0645	Karypis G, Kumar V	A Fast and high-quality multilevel scheme for partitioning irregular graphs
0.0192	Bank RE, Smith RK	The incomplete factorization multigraph algorithm
0.0149	Tang WP, Wan WL	Sparse approximate inverse smoother for multigrid
0.0115	Chan TF, Smith B, Wan WL	An energy-minimizing interpolation for robust multigrid methods
0.0114	Henson VE, Vassilevski PS	Element-free AMGe General algorithms for computing interpolation weights in AMG
0.0108	Hendrickson B, Rothberg E	Improving the Run-time and Quality of Nested Dissection Ordering
0.0092	Karypis G, Kumar V	Parallel multilevel k-way partitioning scheme for irregular graphs
0.0091	Tang WP	Toward an effective sparse approximate inverse preconditioner
0.0085	Saad Y, Zhang J	BILUM Block versions of multielimination and multilevel ILU preconditioner for general sparse linear systems
0.0080	Bridson B, Tang WP	A structural diagnosis of some IC orderings

In the example above, the inner product of the centroids for V. Torczon and V. J. Torczon is 0.98, and thus we would have high confidence that these author names alias a single person (verified by manual inspection of the articles).

As an example use of author disambiguation, we did the following experiment. (1) We selected the top 20 authors of papers in the dataset, i.e., those with the most papers. (2) For each author in the top 20, we selected all papers with any author sharing the same first initial and last name. (3) Next, for each author, we computed the centroid for each set of first initials. (4) We calculated the similarity scores for the centroids. (5) Finally, we manually checked to see which matches were correct.

Table 10 shows the results. We show just the top ten authors before and after disambiguation. The lowest "true match" disambiguation score was 0.4399, and this correctly adjusted the article counts for four authors: T. F. Chan, T. A. Manteuffel, S. F. McCormick and G. H. Golub. All matches above this threshold were true matches; in contrast, the next highest disambiguation score was 0.0964. For three of the above four authors, it was just a matter of connecting authors whose second initial was omitted. However, disambiguating T. F. Chan was less straightforward, as T. Chan and T. M. Chan are also authors in the data. However, the disambiguation scores correctly identify T. Chan as T. F. Chan: T. Chan—T. F. Chan = 0.7935, T. Chan—T. M. Chan = 0.001, and T. F. Chan—T. M. Chan = 0.001. Such differentiation may be helpful when searching for a particular author's entire body of work.

In our initial disambiguation scoring, we aimed at matching centroids to determine author aliasing. However, some of the authors with single initials were not the same author (this can happen with two initials as well, but is less likely). This leads to a further challenge of disambiguating authors. To resolve this problem, we computed disambiguation scores using centroids for authors with two or more initials and individual documents for authors with only a single initial. An example of where this helped correctly determine multiple aliases is Z. Wu. Table 11 lists the papers by these authors. We consider two cases: treating Z. Wu as a single author and taking the centroid of the two papers or treating each paper as separate. In Table 12, Z. Wu, as the author of two papers, appears most similar to author 3. When we separate the articles of Z. Wu and recompute the

Table 10. Authors with most papers before and after disambiguation (threshhold = 0.499)

Before Disambuguation		*After Disambuguation*	
Papers	*Author*	*Papers*	*Author*
17	Du Q	17	Du Q
15	Kunisch K	16	Chan TF
15	Zwick U	16	Manteuffel TA
14	Chan TF	16	McCormick SF
13	Klar A	15	Kunisch K
13	Manteuffel TA	15	Zwick U
13	McCormick SF	13	Klar A
13	Motwani R	13	Golub GH
12	Golub GH	13	Motwani R
12	Kao MY	12	Kao MY

Table 11. Disambiguation of author Z. Wu.

ID	*Author*	*Title(s)*
1a	Wu Z (Zhen)	Fully coupled forward-backward stochastic differential equations and applications to optimal control
1b	Wu Z (Zili)	Sufficient conditions for error bounds
2	Wu ZJ (Zhijun)	A fast newton algorithm for entropy maximization in phase determination
3	Wu ZL (Zili)	First-order and second-order conditions for error bounds
3	Wu ZL (Zili)	Weak sharp solutions of variational inequalities in Hilbert spaces
4	Wu ZN (Zi-Niu)	Steady and unsteady shock waves on overlapping grids
4	Wu ZN (Zi-Niu)	Efficient parallel algorithms for parabolic problems

scores, there is much stronger evidence that authors 1b and 3 are the same author and that author 1a is most likely not an alias for one of the other authors; see Table 13.

Manual inspection of all the articles by this group of authors indicates that authors 1b and 3 are in fact the same person, Zili Wu, and that author 1a is not an alias of any other author in this group. The verified full name of each author is listed in parentheses in Table 11.

5.5 Journal Prediction via Ensembles of Trees

As another analysis approach, we investigated supervised machine learning with the PARAFAC decomposition data as a means of predicting which journal published which paper. Using centroids to represent each journal, as was done for topics or authors in §5.3, did not yield good results because the centroids were not sufficiently distinct.

Instead, we used decision trees and ensembles. Our feature vectors were based on the authority matrix $\mathbf{A}$ from a PARAFAC decomposition with $R = 30$. Consequently, each document was represented by a length-30 feature vector, and the name of the journal which published it as the label value. We split the 5022 labeled feature vectors into 10 disjoint partitions, stratified so that the relative proportion of each journal's papers remained constant across the partitions. Using OpenDT

Table 12. Combined author 1 in disambiguation of Z. Wu.

	1	2	3	4
1	1.00	0.18	0.79	0.03
2	0.18	1.00	0.06	0.06
3	0.79	0.06	1.00	0.01
4	0.03	0.06	0.01	1.00

Table 13. Separated author 1 for disambiguation of Z. Wu.

	1a	1b	2	3	4
1a	1.00	0.01	0.21	0.03	0.07
1b	0.01	1.00	0.09	**0.90**	0.00
2	0.21	0.09	1.00	0.06	0.06
3	0.03	**0.90**	0.06	1.00	0.01
4	0.07	0.00	0.06	0.01	1.00

[3], we performed a 10-fold cross validation of bagged ensembles [11] of C4.5 decision trees. We used an ensemble size of 100; larger ensembles did not improve performance.

Table 14. Journal prediction confusion matrix

		Predicted Journal												
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*Total*
1	SIAM J APPL DYN SYST	0	14	4	1	1	4	0	3	1	1	2	1	32
2	SIAM J APPL MATH	1	318	19	46	3	54	13	31	7	41	12	3	548
3	SIAM J COMPUT	0	29	303	24	29	5	15	8	7	10	109	1	540
4	SIAM J CONTROL OPTIM	0	57	21	346	2	34	20	12	51	22	11	1	577
5	SIAM J DISCRETE MATH	0	12	122	9	40	4	15	2	1	2	53	0	260
6	SIAM J MATH ANAL	0	120	19	56	1	108	15	58	3	34	5	1	420
7	SIAM J MATRIX ANAL A	0	23	11	22	5	8	235	18	18	81	2	0	423
8	SIAM J NUMER ANAL	0	56	13	47	0	37	37	304	13	98	5	1	611
9	SIAM J OPTIMIZ	0	10	19	55	1	4	10	5	228	1	10	1	344
10	SIAM J SCI COMPUT	0	77	7	32	0	36	98	135	23	237	7	4	656
11	SIAM PROC S	0	37	176	21	34	12	9	8	7	13	149	3	469
12	SIAM REV	1	48	13	12	2	13	16	6	6	10	8	7	142

The average accuracy across the folds was 45.3%. The overall confusion matrix (obtained by element-wise summing of the per-fold confusion matrices) is in Table 14. A few caveats are in order before discussion of the results. The journal "SIAM PROC S" (#11) is not actually a journal but rather conference proceedings about a variety of topics. Similarly, the journal "SIAM REV" (#12) contains articles on many different topics. Thus, it is no surprise that the many of the incorrect predictions involve these two publications. It is also important to note that both "SIAM J APPL DYN SYST" (#1) and "SIAM REV" (#12) have very few articles overall. Of the remaining journals, the majority of articles are correctly assigned. Moreover, the journals that seem to often be confused are those that are, in fact, quite similar, such as "SIAM J APPL MATH" (#2) and "SIAM J MATH ANAL" (#6). Moreover, note that "SIAM J SCI COMPUT" (#10) is confused more than the others, which seems reasonable since it is a more diffuse journal than some of the others.

6 Conclusions & Future Work

We represent multiple similarities between documents in a collection as an $N \times N \times K$ tensor and decompose the tensor using PARAFAC. How to best choose the weights of the entries of the tensor is an open topic of research—ours were chosen heuristically.

Different factors from the PARAFAC decomposition are shown to emphasize different link types; see §5.1. Moreover, the highest scoring hubs and authorities in each factor are an interrelated community. The component hub and authority matrices ($\mathbf{H}$ and $\mathbf{A}$) of the PARAFAC decomposition can be used to derive feature vectors for latent similarity scores. However, the rank (R) of the PARAFAC decomposition strongly influenced the quality of the matches; see §5.2. The choice of rank (R) and exactly how to use the component matrices are open questions, including how to combine the hub and authority matrices, how to weight or normalize the features, and whether or not to incorporate the factor weightings, i.e., $\boldsymbol{\lambda}$.

This brings us to two disadvantages of the PARAFAC model. First, the factor matrices are not orthogonal, in contrast to the matrix SVD. A possible remedy for this is to instead consider the TUCKER decomposition [31], which produces orthogonal component matrices and, moreover, can have a different rank for each component. Second, the decomposition of rank R is not the same as the first R factors of the decomposition of rank $S > R$, again in contrast to the SVD [21]. This means that we cannot just overestimate and determine the optimal R by trial-and-error without great expense.

We used the centroids of hub and of authority feature vectors to represent a small body of work (e.g., all the papers with the phrase "GMRES") in order to find related works. As expected, the hub and authority feature vectors produce noticeably different answers, either one of which may be more or less useful in different contexts; see §5.3.

An application of the similarity analysis is in author disambiguation, where we compare centroids to predict which authors with similar names are actually the same. The technique is applied to the subset of authors with the most papers authored in the entire dataset and affects the counts for the most published authors; see §5.4. In future work, we will consider the appropriate choice of the rank (R) for disambiguation, how to choose the disambiguation similarity threshold, and do a comparison to other approaches.

Using the derived authority vectors, we predict which journal each article was published in; see §5.5. Though the accuracy was relatively low, closer inspection of the data yielded clues as to why. For example, two of the publications were not focused publications. Overall, the results revealed similarities between the different journals.

We also plan to revisit the representation of the data on two fronts. First, we wish to add authors as nodes. Hendrickson [18] observes that term-by-document matrices can be expanded to be (term *plus* document)-by-(term *plus* document) matrices so that term-term and document-document connections can be additionally encoded. Therefore, we intend to use a (document *plus* author) dimension so that we can explicitly capture connections between documents and authors as well as the implicit connections between authors, e.g., colleagues, conference co-organizers, etc. Second, in order to make predictions or analyze trends over time, we intend to incorporate transient information using an additional dimension for time.

7 Acknowledgments

Data used in this paper was extracted from the Science Citation Index Expanded, Thomson ISI, Philadelphia, PA, USA. We gratefully acknowledge **Brett Bader** for his work on the MATLAB

sparse tensor toolbox which was used in our computations, and **the TMG Toolbox creators** for providing a helpful tool for generating term-document matrices in MATLAB (see [33]).

References

[1] E. Acar, S. A. Çamtepe, M. S. Krishnamoorthy, and B. Yener. Modeling and multiway analysis of chatroom tensors. In *ISI 2005: IEEE International Conference on Intelligence and Security Informatics*, Lecture Notes in Computer Science 3495, pp. 256–268. Springer Verlag, 2005.

[2] L. A. Adamic and E. Adar. How to search a social network. Technical report, HP Laboratories, Palo Alto, CA, Oct. 2004.

[3] R. Banfield et al. OpenDT Web Page. `http://opendt.sourceforge.net/`, 2004.

[4] A. Barábasi, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *Physica A*, 311(3–4):590–614, 2002.

[5] R. Bekkerman and A. McCallum. Disambiguating web appearances of people in a social network. In *WWW 2005: Proceedings of the 14th international conference on World Wide Web*, pp. 463–470. ACM Press, 2005.

[6] M. W. Berry, S. T. Dumais, and G. W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Rev.*, 37(4):573–595, 1995.

[7] K. W. Boyack. Mapping knowledge domains: characterizing PNAS. *P. Natl. Acad. Sci.*, 101(Suppl. 1):5192–5199, Apr. 6 2004.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *WWW7*, pp. 107–117. Elsevier, 1998.

[9] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition. *Psychometrika*, 35:283–319, 1970.

[10] S. de Lin and H. Chalupsky. Using unsupervised link discovery methods to find interesting facts and connections in a bibliography dataset. *ACM SIGKDD Explor. Newsl.*, 5(2):173–178, 2003.

[11] T. Dietterich. Ensemble methods in machine learning. In J. Kittler and F. Roli, editors, *First International Workshop on Multiple Classifier Systems*, number 1857 in Lecture Notes in Computer Science, pp. 1–15. Springer-Verlag, 2000.

[12] S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman. Using latent semantic analysis to improve access to textual information. In *CHI '88: Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 281–285. ACM Press, 1988.

[13] N. K. M. Faber, R. Bro, and P. K. Hopke. Recent developments in CANDECOMP/PARAFAC algorithms: a critical review. *Chemometr. Intell. Lab.*, 65(1):119–137, Jan. 2003.

[14] J. Gehrke, P. Ginsparg, and J. Kleinberg. Overview of the 2003 KDD cup. *ACM SIGKDD Explor. Newsl.*, 5(2):149–151, 2003.

[15] L. Getoor and C. P. Diehl. Link mining: a survey. *ACM SIGKDD Explor. Newsl.*, 7(2):3–12, 2005.

[16] L. Getoor, N. Friedman, D. Koller, and B. Taskar. Learning probabilistic models of link structure. *J. Mach. Learn. Res.*, 3:679–707, 2003.

[17] R. A. Harshman. Foundations of the PARAFAC procedure: models and conditions for an "explanatory" multi-modal factor analysis. *UCLA working papers in phonetics*, 16:1–84, 1970.

[18] B. Hendrickson. Latent semantic analysis and Fiedler retrieval. Submitted for publication to Linear Algebra and Applications. Earlier version in Proc. Text Mining 2006, 2006.

[19] S. Hill and F. Provost. The myth of the double-blind review?: author identification using only citations. *ACM SIGKDD Explor. Newsl.*, 5(2):179–184, 2003.

[20] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[21] T. G. Kolda. Orthogonal tensor decompositions. *SIAM J. Matrix Anal. A.*, 23(1):243–255, 2001.

[22] T. G. Kolda and B. W. Bader. The TOPHITS model for higher-order web link analysis. In *Workshop on Link Analysis, Counterterrorism and Security*, 2006.

[23] T. G. Kolda, B. W. Bader, and J. P. Kenny. Higher-order web link analysis using multilinear algebra. In *ICDM 2005: Proceedings of the 5th IEEE International Conference on Data Mining*, pp. 242–249. IEEE Computer Society, 2005.

[24] N. Liu, B. Zhang, J. Yan, Z. Chen, W. Liu, F. Bai, and L. Chien. Text representation: From vector to tensor. In *ICDM 2005: Proceedings of the 5th IEEE International Conference on Data Mining*, pp. 725–728. IEEE Computer Society, 2005.

[25] A. McGovern, L. Friedland, M. Hay, B. Gallagher, A. Fast, J. Neville, and D. Jensen. Exploiting relational structure to understand publication patterns in high-energy physics. *ACM SIGKDD Explor. Newsl.*, 5(2):165–172, 2003.

[26] M. J. Rattigan and D. Jensen. The case for anomalous link discovery. *ACM SIGKDD Explor. Newsl.*, 7(2):41–47, 2005.

[27] A. Smilde, R. Bro, and P. Geladi. *Multi-way analysis: applications in the chemical sciences*. Wiley, 2004.

[28] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. CubeSVD: a novel approach to personalized Web search. In *WWW 2005: Proceedings of the 14th international conference on World Wide Web*, pp. 382–390, 2005.

[29] D. Tao, X. Li, W. Hu, S. Maybank, and X. Wu. Supervised tensor learning. In *ICDM 2005: Proceedings of the 5th IEEE International Conference on Data Mining*, pp. 450–457. IEEE Computer Society, 2005.

[30] G. Tomasi and R. Bro. PARAFAC and missing values. *Chemometr. Intell. Lab.*, 75(2):163–180, Feb. 2005.

[31] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[32] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: TensorFaces. In *ECCV 2002: 7th European Conference on Computer Vision*, Lecture Notes in Computer Science 2350, pp. 447–460. Springer-Verlag, 2002.

[33] D. Zeimpekis and E. Gallopoulos. TMG: A MATLAB toolbox for generating term-document matrices from text collections. In J. Kogan, C. Nicholas, and M. Teboulle, editors, *Grouping Multidimensional Data: Recent Advances in Clustering*, pp. 187–210. Springer, 2006.

DISTRIBUTION:

2 MS 9018
Central Technical Files, 8945-1

2 MS 0899
Technical Library, 4536

1 MS 0188
D. Chavez, LDRD Office, 1011