WordSpace — Visual Summary of Text Corpora

Ulrik Brandes, Martin Hoefer, Jürgen Lerner

Dept. of Computer & Information Science, Konstanz Univ.,Box D 67, 78457 Konstanz, Germany

ABSTRACT

In recent years several well-known approaches to visualize the topical structure of a document collection have been proposed. Most of them feature spectral analysis of a term-document matrix with influence values and dimensionality reduction. We generalize this approach by arguing that there are many reasonable ways to project the term-document matrix into low-dimensional space in which different features of the corpus are emphasized. Our main tool is a continuous generalization of adjacency-respecting partitions called structural similarity. In this way we obtain a generic framework in which influence weights in the term-document matrix, dimensionality-reducing projections, and the display of a target subspace may be varied according to nature of the text corpus.

Keywords: Information Retrieval, Text Visualization, Data Analysis, Graph Drawing

1. INTRODUCTION

The Internet is the prime example of ever-growing information repositories that are mostly in textual form. With increasing amounts of data it becomes more and more difficult for users to derive material of interest, to search efficiently for specific content or to gain an overview of influential, important and relevant material. During the last decade there have been increased research efforts in the area of information retrieval to foster efficient search and clustering of documents. These approaches have been purely analytical, and there have been very few attemps to translate the results into easily accessible visual form. In this paper we deal with the problem to provide and overview over important notions, topics and themes, which are covered by a large collection of documents. We will propose a framework to identify similarities between documents and words and to retrieve the underlying influence and importance of topics, words and documents in a corpus. Our approach is graph-theoretic and identifies similar structures in a term-document graph using spectral analysis. The framework used for content analysis can alternatively be considered as a generalization of latent semantic indexing, a well-established method for information retrieval. Additionally, we incorporate graph drawing methods to translate our results into intuitively understandable visual layouts, thereby communicating complex analytical information through a well-designed visualization.

The remainder of this chapter will provide information on related work and mathematical preliminaries. The following sections deal with the three stages of WordSpace. Sect. 2 will introduce text analysis techniques to measure word influence in documents. In Sect. 3 a framework is presented for spectral content analysis and detection of structural similarities. Sect. 4 explains the transformation into a graphical layout. Finally, in Sect. 5 we report test results of the procedure and performance observations on a document collection of research articles coming from social network analysis.

Related Work The analysis of texts using networks and graph theory is receiving increasing attention.[1] There are recent attempts to display the evolution of dynamic discourse by using standard graph-drawing techniques to represent text structure.[2] Here the focus is solely on graph drawing methodology and does not incorporate similarities or semantic meanings of words into the layout. The approach is based on text representation using Centering Resonance Analysis (CRA),[3] which will be of use in our approach as well (see Sect. 2 for an introduction).

Recently, several approaches have appeared in the visualization literature to display and explore large collections of documents. Some of them like InfoSky,[4] the Hyperbolic Browser[5] or Information Pyramids[6] employ existing hierachical structuring of the document corpus. Our approach, however, is a general integrated procedure for both analysis and visualization of documents and major topics and therefore more similar to systems such as Bead[7,8] and SPIRE.[9,10] Both Bead and SPIRE adopt what is called the ecological approach[9] by imitating a natural environment to visualize documents.

E-Mail: {brandes|hoefer|lerner}@inf.uni-konstanz.de

Bead constructs a landscape-like layout, in which similarity is communicated by proximity using word frequency measures for influence detection and an iterative force-based layout algorithm.[11] The height of a document in the landscape is determined by the number and importance of terms the document contains. Important words are dynamically placed in the visualization depending on the viewpoint of the user and frequency of word occurrence in the current field of view.
SPIRE is a more advanced system, which applies a specialized text analysis method to reduce the number of relevant and discriminative terms. Hierachical clustering or k-means clustering are used to generate a clustering of documents in high-dimensional vector space. To generate a visualization cluster centroids are placed into the plane and documents are positioned in order to preserve the distances to cluster centroids. In this step spectral methods or an iterative stress minimization algorithm are used for smaller or larger document collections, respectively. Finally, SPIRE includes two layout schemes for exploring the collection. Either the document collection can be displayed in a galaxy layout, or important topics and document clusters can be identified with a landscape-like layout similar to the Bead system. The SPIRE system is implemented in the commercial IN-SPIRE tool.[12]

Bead as well as SPIRE consist of a variety of heuristic methods that interplay in generating a visualization. WordSpace rather offers a simple, mathematically rigorous, unified approach to the visualization of both terms and documents. It uses a cumulative, powerful mathematical analysis for measuring similarities, constructing a high-dimensional layout and transferring important properties to a two-dimensional layout. Hence, WordSpace is a more suitable, well-defined analytical procedure for extracting all relevant information from a meaningful mathematical text representation.

As WordSpace uses spectral analysis and low-rank approximations of matrices to analyze, cluster and visualize data, it is related to a vast amount of research that has been conducted in this area throughout the last decades. For our purposes, especially the work on Latent Semantic Indexing (LSI),[13,14] a well-established technique for information retrieval, is of interest.

Preliminaries

In the following we assume that the reader is familiar with basic facts and notions from graph theory and linear algebra. We will denote the transpose of a matrix A as A^T. An important construct that will be key to our approach is the *singular value decomposition*, which for a real matrix A is given by three real matrices U,V,Σ such that $A = U\Sigma V^T$, where Σ is a diagonal matrix. A column u_i (row v_i) is called the ith *left (right) singular vector* and the element σ_i of Σ the corresponding *singular value*. For the special case of a symmetric matrix, this notion is equal to the *eigen decomposition* of A. It is given by two symmetric real matrices V,Λ such that $A = V\Lambda V^T$, where Λ is a diagonal matrix. A column v_i of V is called the i-th *eigenvector* of A and the corresponding element λ_i of Λ the corresponding *eigenvalue*. Note that the singular values of A are the square roots of the eigenvalues of A^TA and AA^T.

For the remainder of the paper we will assume that there is a collection D of texts or documents, from which we derive a set W of words or terms. If we consider either word or document, we speak of an item. The collection has $w = |W|$ words, $d = |D|$ documents and $n = w + d$ items.

2. REPRESENTING WORD INFLUENCE IN A DOCUMENT CORPUS

In this section we discuss the translation of D into an abstract mathematical formulation, which in the later stages is used to build a meaningful visualization of the important topics and phrases. We will use text analysis methods to derive a term-document matrix indicating the relevance of terms in respective documents. Using this matrix we construct a weighted bipartite graph between nodes corresponding to words and documents. This graph is further analyzed and visualized in the following sections.

Text Analysis In the field of text analysis several methods have been developed to determine the most important topics and phrases of a document. A recent variant is *Centering Resonance Analysis* (CRA),[3] a method of network text analysis.[1] For our purposes CRA has distinct advantages over other approaches that could be applied. It is a representational method not relying on context-specific semantic rules, training sets, or predefined document collections. CRA produces stand-alone, abstract representations of texts, which can be analyzed separately or combined and compared with other CRA representations.
For a grammatically correct text T of the document collection CRA builds a graph $G(T) = (V(T), E(T))$ as follows. Initially, noun phrases are extracted from each sentence. The words of the noun phrases are nouns and adjectives, which form

the set $V(T)$ such that each vertex corresponds to a word occurring in T. Two vertices are connected if the corresponding words co-occur in the same noun phrases or occur on adjacent ends of consecutive phrases within a sentence. After processing all sentences of a text, a network of words is formed by accumulating the weight of parallel edges. It represents subjects and objects of the text and the strength of their relations. The graph $G(T)$ is called *CRA network* of the text T.
Elaborate texts contain important words at strategical positions to support a coherent and meaningful message. CRA uses network centrality to assess whether words are located at influential positions in the text. Specifically, centrality measures capture the extend to which a word is involved in association chains between other words in the network. CRA has commonly been combined with *betweenness centrality*,[15] $c_B(v)$, of a vertex $v \in V(T)$, which is defined as

$$c_B(v) = \sum_{s \neq v \neq t \in V(T)} \frac{\tau_G(s,t|v)}{\tau_G(s,t)}$$

where $\tau_G(s,t)$ and $\tau_G(s,t|v)$ are the number of shortest paths between s and t and those passing through v, respectively. As the edge weight does not denote the length but rather the number of parallel connections, shortest paths are calculated assuming an edge cost of 1. In τ_G a path then accounts for the number of edge-disjoint connections, i.e. the product of the weights of its edges. A word with high betweenness centrality is therefore influential because it is heavily involved in channeling flows of meaning in the network. A criticism to betweenness centrality has been the fact that only shortest paths are considered ignoring other relatively short connections in the network. Hence, to capture the structural position of a word in the CRA network more accurately, we will alternatively use *current-flow betweenness centrality*.[16,17] The graph $G(T)$ is assumed to form an electric network, and edge weights correspond to conductance or inverse resistance. The centrality $c_{CB}(v)$ for $v \in V(T)$ is defined as

$$c_{CB}(v) = \sum_{s \neq v \neq t \in V(T)} \tau^c_G(s,t|v)$$

where $\tau^c_G(s,t|v)$ is the fraction of current that runs through an inner vertex v if a unit of current flows between vertices s and t.

As a third measure we included a version of the *term frequency*. After nouns and adjectives are extracted from the documents we simply measure influence of a word w_i in a document d_j as being the normalized number

$$\frac{cnt(w_i,d_j)}{\sum_{w_k} cnt(w_k,d_j)},$$

where $cnt(w_i,d_j)$ is the number of occurrences of word w_i in document d_j. The normalization with the *sum of word occurrences* in document d_j is done analogously to the centrality measures in order to rule out document length as influence factor.

Finally, we experimented using TFIDF, which is a standard frequency-based measure in information retrieval.[18] For a document the *inverse document frequency* of word w_i is defined as

$$idf_i = \log\left(\frac{|D|}{n_i}\right),$$

where n_i is the number of documents containing w_i. This measure is multiplied with a more commonly used notion of *term frequency*

$$tf_{ij} = \frac{cnt(w_i,d_j)}{max_i\, cnt(w_i,d_j),}$$

hence the number of occurrences of w_i in d_j, which is in this case normalized by the *maximum number of occurrences* of any word in d_j. The influence value of word w_i in document d_j is then given by

$$tf_{ij} * idf_i.$$

This measure tries to capture the intuition that a word occurring in document d_j more often than in other documents is influential for the unique content of d_j. Hence, words appearing in d_j more often than on average in the other documents of the collection are assigned a high weight. If a word occurs e.g. in any document, its weight will be 0. Notice that this is just the opposite understanding of word influence than with centrality or term frequency. The resulting effects on the displayed information will be discussed in Sect. 5.

Term-Document Graph For each document T the text analysis procedure provides a vector of influence values of the nouns and adjectives. Analogous to standard clustering and retrieval methods for text collections[13, 14] we combine the influence values for the documents in the collection and construct a term-document matrix $A \in \mathbf{R}^{\mathrm{w}\times\mathrm{d}}$. Entry a_{ij} is set to the influence value of term i in document j if it is present and to 0 otherwise. A is used to define the undirected weighted bipartite term-document graph $G = (V,E)$ with $V = W \cup D$. The weighted symmetric adjacency matrix is given by

$$B := \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} .$$

3. IDENTIFYING KEYWORDS AND THEMES

Just like information retrieval (IR), visualization of text faces the problem of synonymy. Synonymy refers to the fact that different words (like *car* and *automobile*) may have the same meaning. A dual problem is that different documents may treat a similar topic, even if they have only very few words in common. In a good visualization, we wish to place synonymous words and documents with similar content close to each other.
Another problem that has to be solved is that of determining the importance of words. Even small collections of texts are likely to contain thousands of different words. For larger bodies of text, this number goes easily up to hundreds of thousands. Displaying all words equally sized yields a very confusing image in which the user will hardly be able to recognize valuable information. This problem can be overcome by displaying important words in larger text-size, thus giving a coarse overview of the most important topics and providing the information in which areas it is worth to zoom in. Synonymy and importance of words will be addressed in this section. We start by considering the first problem. An optimal solution to the synonymy-problem would partition the set of words and the set of documents such that two words (documents) are in the same class if and only if they are synonyms (semantically equivalent). One class could then be seen as representing a topic, theme, or concept. However, in practice two words will never have precisely the same meaning, and there is some randomness in word usage. A solution to get around this problem is to use a relaxation of the boolean formulation—putting a word (document) into one class or the other—to the continuous notion of assigning real-valued degrees of class-membership to words (documents). These degrees of membership are the importance that the word (document) has for a specific topic. The importance of a word (document) for the whole text collection is obtained by summing up the importance over all topics. Thus, the continuous assignment of topics to terms and documents provides solutions to the synonymy and to the importance problem.

In this section we consider the second stage of WordSpace and sketch a framework for the continuous assignment of topics to terms and documents. Based on the vector space corresponding to the term-document matrix A we make topic-assignments depending on the structure of the space that yield meaningful topics. We characterize the assignments that satisfy certain meaningful conditions and show how to compute them. It turns out that latent semantic indexing (LSI), which is an established method for document retrieval, is a one of several choices in our framework.

3.1. Term-Document Spaces

Given a term-document graph $G = (W \cup D, E)$, defined in Sect. 2, we derive the continuous notion of a *term-document space* $\mathcal{G}$, which will be the *graph space* of G.[19] In a term-document space it is possible to express not only that a word i is relevant to a certain degree for a document j (as expressed by the weight of the edge (i, j) in graph G), but also that a *weighted subset* of words is contained in a *weighted subset* of documents. To be precise, we understand by a *weighted subset* of words, a real-valued function $f: W \to \mathbf{R}$ defined on the set of words W. The *degree of membership* of word i to the weighted subset f is the real number $f(i)$. The set of all weighted subsets of words $\mathcal{W}(G) = \{f: W \to \mathbf{R}\}$ is a real vectorspace. A single word $i \in W$ is identified with the special element of $\mathcal{W}$ that maps i to 1 and all other words to 0. Similarly, we define $\mathcal{D}(G) = \{f: D \to \mathbf{R}\}$ the space of all weighted subsets of documents. Together, we obtain the vertexspace $\mathcal{V} = \mathcal{W} \oplus \mathcal{D}$ induced by the vertex set $V = W \cup D$ of G.
The matrix B, defined in Sect. 2, maps a document to the terms it contains and a term to the documents it is contained in. Since terms and documents together form a basis of $\mathcal{V}$, this mapping naturally extends to a linear mapping on weighted subsets of terms and documents. The degree to which a weighted subset of words $f_w \in \mathcal{W}$ is contained in a weighted subset of documents $f_d \in \mathcal{D}$ is defined as $\langle f_d, B(f_w)\rangle$, where $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. If f_w and f_d happen to be a single word, resp. document, this product is the weight of edge (f_w, f_d) in G.

3.2. Projections of Term-Document Spaces

Next, we define *projections* as a relaxation of partitions of words and documents. Assume, there are k different topics in the collection of texts. A discrete partition would partition words into k classes of words and documents into k classes of documents, one for each topic. This assignment is very unlikely to exist, and a projection relaxes it to real degrees of membership. A projection P defines for each class of words i and each word j a real number p_{ij}, which is the degree of membership of word j to class i. Similarly, P defines for each class of documents i' and each document j' a real number $p_{i'j'}$. Reasonable conditions are that the different classes are orthogonal and normalized, and that classes of words are disjoint to classes of documents. These remarks are formalized in the next definition.

DEFINITION 3.1. *A* projection *(of term-document spaces) is a real matrix P of dimension $2k \times n$, satisfying*

1. *$PP^T = I$ (i. e., P has orthonormal rows) and*
2. *$P(\mathcal{W})$ is orthogonal to $P(\mathcal{D})$ (i. e., P projects words and documents onto disjoint classes).*

The second condition is equivalent to the condition that P consists of a matrix P_W, which maps words to classes of words, and a matrix P_D which maps documents to classes of documents:

$$P = \begin{bmatrix} P_W & 0 \\ 0 & P_D \end{bmatrix} .$$

A discrete partition of the set of words (documents) induces an equivalence relation, where two words (documents) are equivalent if they are in the same class. Two documents are semantically equivalent (synonyms), if they represent the same topic. Projections induce relaxations of equivalence relations, which will be called *similarities* in the following. The similarity $s(i, j)$ of two words i and j is defined by $s(i, j) := \sum_{l=1}^{2k} p_{li} \cdot p_{lj}$, i. e., similarity is high if i and j belong to a high degree to the same classes. The similarity of two documents is defined accordingly.

DEFINITION 3.2. *Let P be a projection. The matrix S that contains the similarity values of all pairs of words and documents is called the* similarity *associated to P, i. e.,*

$$S := P^T P \ .$$

The second condition on projections ensures that a similarity distinguishes between words and documents, i. e., a word and a document have always similarity zero to each other. For each similarity there is unique associated projection (up to isomorphism) and vice versa.[19] Projections reduce the complexity of term-document spaces by classifying words and documents. Next, we show that the reduced space (or class space) inherits the structure of the initial term-document space such that classes of documents contain classes of words.
We define the reduced space $\bar{\mathcal{V}} := P(\mathcal{V})$, the space of word classes $\bar{\mathcal{W}} := P(\mathcal{W})$ and the space of document classes $\bar{\mathcal{D}} := P(\mathcal{D})$. The degree to which a class of words $C_w \in \bar{\mathcal{W}}$ is contained in a class of documents $C_d \in \bar{\mathcal{D}}$ is defined as the average degree to which a word belonging to C_w is contained in a document belonging to C_d. The following definition meets this goal.[19, 20]

DEFINITION 3.3. *Let $\mathcal{G} = (\mathcal{V}, B)$ be a term-document space and P a projection. The* quotient *of $\mathcal{G}$ over P is the term document space $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{B})$, where $\bar{B}$ is defined by $\bar{B} := PBP^T$.*

3.3. Structural Projections

In this subsection we consider a condition which ensures that projections yield meaningful topics. It states that words (documents) regarded as equivalent must be contained in (contain) the same number of equivalent documents (words).

To formalize this we will use the notion of *structural* similarities (c.f.[19]).

DEFINITION 3.4. *Let $(\mathcal{V}, B)$ be a term-document space, P a projection, and $\bar{B} = PBP^T$ the adjacency matrix of the quotient space. Then, P (and its associated similarity) is called* structural *if $PB = \bar{B}P$.* The equivalent conditions for a projection (or similarity) will allow to derive further desirable properties and yield efficient algorithms to compute structural projections. Let A be the term-document matrix, defined in Sect. 2, and $\sigma_1, \ldots, \sigma_n$ be the nonzero singular values of A to the (left and right) singular vectors u_i, v_i, then the nonzero eigenvalues of A are $\pm\sigma_1, \ldots, \pm\sigma_n$ to the eigenvectors

$$\begin{pmatrix} u_i \\ \pm v_i \end{pmatrix} .$$

We call a subspace $\mathcal{U} \subseteq \mathcal{V}$ *consistent* with the matrix B of a term-document space if $\mathcal{U}$ is generated by eigenvectors of B and has the property that if $v \in \mathcal{U}$ is an eigenvector to the eigenvalue λ then there is a $v' \in \mathcal{U}$ which is an eigenvector to the eigenvalue $-\lambda$.

THEOREM 3.5. *Let $(\mathcal{V}, B)$ be a term-document space, P a projection, and S the associated similarity. The following assertions are equivalent.*

1. *P is structural;*
2. *$SB = BS$;*
3. *S is the orthogonal projection onto a subspace consistent with B.*

Proof. The condition on consistent subspaces is necessary and sufficient that a structural P does not project a term and a document on the same class. The other assertions follow with linear algebra arguments.[19] □

The above theorem implies that structural projections of a term-document space correspond one-to-one to subsets $\Lambda' \subseteq \Lambda$ of the spectrum Λ of B, with the property

$$\forall \lambda: \lambda \in \Lambda' \Leftrightarrow -\lambda \in \Lambda' \quad . \tag{1}$$

3.4. Comparison of LSI and Structural Projections

LSI is an established method in document retrieval. Using the term-document matrix A characterized in Sect. 2 one computes the singular value decomposition of A. Then, all but the k largest singular values are set to zero, which yields a matrix A_k as an approximation of A. It follows from Theorem 3.5 that this method is equivalent to the structural projection onto the subspace spanned by the eigenvectors corresponding to the k largest (positive) and the k smallest (negative) eigenvalues of the matrix B (defined in Sect. 2). Thus, standard LSI methods are a restricted usage of structural projections. Structural projections are more general in allowing other (than the largest) eigenvalues to be kept. While the specific method of taking the largest eigenvalues has some advantages,[21] in some cases the best projections are omitted by LSI but included in the set of structural projections. LSI shows a failure whenever the set of words and documents can be partitioned into dense clusters, such that there are very few (or only weak) edges between clusters, but synonymous words are scattered over the clusters. Examples, where this situation arises naturally, are multilingual text collections, or heterogenous text collections written by groups of authors with almost disjoint vocabulary.

3.5. Choosing Projections

The property of being structural assures that a projection yields classes of words and documents such that equivalent words are contained in the same number of equivalent documents and vice versa. However, there is in general a huge number of structural projections for a given term-document space. Although we have no general applicable mechanism that is guaranteed to yield in all cases the best projection, several heuristic methods can be applied, as will be sketched here. Recall that choosing a specific structural projection is equivalent to choosing a subset $\Lambda' \subseteq \Lambda$ of the spectrum Λ of B with the property (1).

- A first possibility would be to adopt the method of LSI and to choose the k pairs of eigenvalues $\pm\lambda$ with the largest absolute value. However, we have pointed out above that in some cases LSI fails to find the best projection.
- A second motivation for the choice of Λ' is that of maximizing the stability of the projection. In many scenarios it seems to be desirable that small changes (like adding or deleting a small number of documents, or choosing a slightly different method for the preprocessing of texts) should not change the projection (and thus, the displayed image of the text) too much. Stability theory for eigenspaces (see, e. g.,[22]) suggests that the projections are stable if the minimal distance between eigenvalues in Λ' and eigenvalues in $\Lambda \setminus \Lambda'$ is big. Thus, in order to maximize stability the projection of choice can be found by searching for large gaps in the spectrum and choosing Λ' accordingly.
- A third possibility arises if the quotient $(\bar{\mathcal{V}}, \bar{B})$ of a term-document space $(\mathcal{V}, B)$ is known, or constrained, for example, by an *a priori* knowledge about the topics that should be treated in the given collection of texts. In this case, the subset Λ' (and thus the chosen projection) is determined by the spectrum of $\bar{B}$, which must be equal to Λ' (see[19]).

4. LAYOUT AND VISUALIZATION

In this section we assume that we have given a term-document space $(\mathcal{V}, B)$ and a structural projection P. Based on this we want to find nice and meaningful drawings of the terms and documents. We present three variants to extract a layout from a given projection and its associated subspace.

4.1. Coordinate Generation

Laplacian Spectral Drawing with Projections The first method closely resembles the one of Koren,[23] with the difference that the motivation to his work has been speed-up of computation and not the visualization of similarity.
Following common principles in graph drawing,[24] a desirable goal is to display words close to the documents they are contained in. To avoid the trivial (and undesired) solution of drawing all words and documents in one point, in addition the layout is constrained to use a large portion of the available area, i. e., that words and documents should not be drawn too close to each other. It is well-known[23] that this problem is equivalent to the one of solving

$$\min_{x \in \mathbf{R}^n} \frac{x^T L x}{x^T x} \quad . \tag{2}$$

Here L denotes the *Laplacian* of B, defined by

$$L_{ij} = \begin{cases} deg(i) & i = j \\ -a_{ij} & i \neq j \end{cases} \quad .$$

where $deg(i)$ is the degree of vertex i in G. The best solutions to problem (2) are the eigenvector x and y of L with the lowest positive eigenvalues. They provide the coordinates of terms and documents of a two-dimensional drawing. This is an optimal solution to the problem of displaying words close to their documents, preventing at the same time that the whole graph collapses to a single point.
However, this solution has not yet taken into account that different words may by synonyms and thus should be drawn close to each other. Therefore, we define as the second layout goal that words (documents) with high similarity should be drawn close to each other. Solutions to this problem can be obtained by first finding a good layout for word and document *classes* and then drawing individual words (documents) according to their degrees of membership to the different classes. More formally, assume that we have a two-dimensional layout $\bar{x}, \bar{y} \in \mathbf{R}^{2k}$ for word and document classes. The degrees to which a word w_i is in the different classes are in the i'th column of P. That is the x-coordinate of word w_i can be defined by $x_i := \sum_{l=1}^{2k} p_{li} \cdot \bar{x}_l$. Writing this in terms of matrix multiplication we get a two-dimensional layout $x, y \in \mathbf{R}^n$ for words and documents by

$$x = P^T \bar{x}, \text{ and } y = P^T \bar{y} \quad .$$

This "pull-back" of a layout on the set of classes to a layout on the set of words ensures that words with high degree of similarity (which are almost in the same classes) are drawn close to each other. Now all layouts x, y, that take into account similarity of words and documents, are of the form $x = P^T \bar{x}$, and $y = P^T \bar{y}$ for vectors $\bar{x}, \bar{y} \in \mathbf{R}^{2k}$. Thus, in order to find layouts that minimize edge length, while respecting similarities, we have to solve the new problem (c.f. (2))

$$\min_{x = P^T \bar{x}, \bar{x} \in \mathbf{R}^{2k}} \frac{x^T L x}{x^T x} \text{ , or equivalently } \min_{\bar{x} \in \mathbf{R}^{2k}} \frac{\bar{x}^T \bar{L} \bar{x}}{\bar{x}^T \bar{x}} \quad , \tag{3}$$

where $\bar{L} := PLP^T$ is the reduced Laplace matrix on the set of word and document classes. Hence, as above, we pick as optimal solution to problem (3) the two eigenvectors $\bar{x}$ and $\bar{y}$ of $\bar{L}$ with the lowest positive eigenvalues. Together, the two vectors $x = P^T \bar{x}$ and $y = P^T \bar{y}$ provide optimal coordinates of terms and documents of a two-dimensional drawing, which displays words close to their documents under the condition that synonyms must be treated equally.

Drawing of low-rank-approximated graphs In the previous subsection we applied the projection to matrix L to project the Laplacian into the low-dimensional topic subspace. In this space we minimized the graph layout of topics by picking eigenvectors as coordinate values. Finally, the topic positions were transferred back into the space of words and documents. Here we propose a slightly different approach. Instead of projecting the Laplacian we create the Laplacian of the projected space. First, a low-rank approximation of the term-document space is created, which is characterized by matrix $B' = P^T BP$.

Matrix B' has only rank k, so for each item the corresponding row in B' is only a linear combination of k vectors. Hence, this yields description in terms of the k topics. In B' similar documents and words are then associated with similar vectors. Actually, this is the standard mechanism also used by LSI, where e.g. the angle between the vectors of B' is used to answer relevance queries. To the weighted bipartite graph corresponding to B' we can directly apply graph drawing methods. As an example, we may generate the Laplacian of B' and then use small eigenvectors to get coordinates. Note that in this way projection and layouting are not integrated but rather applied sequentially.

k-cluster Drawing of Projection Space It is known that Laplacian spectral drawing is problematic when applied to irregular graphs. In these cases the optimization criterion tends to place a small number of nodes with small connectivity in the periphery of the drawing. These nodes then account for space usage, while important and highly connected parts of the graph still collapse in the origin to keep overall edge lengths small.
To cope with this problem we propose a circular *k-cluster drawing*, which uses columns of the projection matrix as contributions to the coordinates. Note that a projection can be identified by a set of eigenvalues of B. Actually, a projection matrix P can be generated by picking the eigenvectors of B corresponding to the eigenvalues and using them as columns of P. The projection, however, contains all the relevant similarity and topic class information. Moreover, each vector represents one of the topic classes mentioned in the previous section. Hence, it is natural to use this information directly and calculate coordinates from eigenvectors of B. As we expect that most words can be put to large amounts into one or two topic classes, we can expect that the corresponding entries are only large for one or two eigenvectors.
Our method works as follows. It considers the k eigenvectors of the projection matrix and assigns each one a direction from the offspring. We use a regular partition of the cycle, i.e. the first vector is pointed in positive x-direction, the second at an angle of $360/k$ degrees, the third at $720/k$ etc. For an item the size of the entry in a vector specifies how far the item is moved in the corresponding direction. In the beginning each item is placed in the origin. Then all the moves are sequentially applied, and the item is drawn at the resulting position.
Our method has some distinctive advantages. In most cases important words and documents are important for only one or two topics. These items then have high entries in one or two columns of P and are moved to the border of the drawing. This effectively reduces the amount of clutter produced by Laplacian methods. Furthermore, due to the connection between vectors and topics, *topic slices* evolve—slice parts of the cycle where a certain class of words and documents is dominant.

4.2. Label Size

As indicated at the beginning of Sect. 3, a projection also provides a measure of importance, or centrality for individual words and documents. We visualize this importance by drawing important words and documents using larger font. Formally, given a projection P we define the *importance* or *centrality* of word (document) i by

$$c_i := \sqrt{\sum_{l=1}^{2k} p_{li}^2} \ .$$

This definition favors words that belong mostly to one topic, over words that are distributed over several topics. The larger size of words that are dominant for a topic, enables the user to recognize the topic he is interested in "at a first glimpse" and thus provides quickly the information where it is worth to zoom in.

5. EXAMPLE VISUALIZATION

The WordSpace layout was tested on a document collection of abstracts from social network analysis literature. The collection encompassed abstracts to the 2004 Sunbelt Conference, which is an annually event in social network research. We combined each abstract with the title and author names to represent a single document. This resulted in a collection of 274 documents in total. A total of 5443 words were extracted from these documents. Using centrality and frequency analysis described in Sect. 2 we obtained four matrices A_{SB}, A_{CB}, A_{TF} and A_{IDF} for shortest-path, current-flow betweenness, term frequency and TFIDF, respectively. Suitable projections were identified using the spectrum of the term-document spaces, which can be immidiately derived with the singular values of the A-matrices. Typically term-document matrices representing word influence in linguistic texts have a so-called *low-rank-plus-shift*-structure.[25] The singular value distribution of such a matrix has a high first eigenvalue, decreases rapidly and levels off in the following. However, the singular values are never close to 0 unless the matrix is *rank deficient*, i.e. does not have maximum possible rank.

In Sect. 3.5 we considered motivations for picking good projections. As there is in our example no a-priori indication, how many different topics or concepts underlie our document collection, we applied the criterion of maximum stability. Due to the rapid decrease in the initial eigenvalues, however, the maximum stability is achieved by chosing the projection corresponding to the largest k singular values of A. The constant k has to be chosen appropriately, i.e. representing a good choice between accurarcy and stability of the projection. This, however, is exactly the same choice as made by classic LSI methods. In contrast to our motivation LSI is motivated by deriving the best low-rank approximation of A to restore as much information of A in the projected subspace as possible.[21] In our case picking the largest singular values turns out to also maximize the stability of the projection subject to matrix perturbation. Hence, in the example we identified the differences between consecutive singular values and picked the k largest eigenvalues with the corresponding eigenvectors. The choice of k was also connected to the drawing method. We experimented with all presented variants, however, the k-cluster method yielded best results, which will be presented on the next pages. In order to be able to distinguish important topics more easily we chose a relatively small $k = 8$ (see Fig. 1). In the previous sections it was already discussed that the WordSpace layout is most suitable for identifying structure and similarities in the document collection. The TFIDF measure can be expected to produce quite different results due to its nature of strengthening unique terms and notions of single documents. Similarity is (at least on a syntactic level) completely suppressed, which alters the meaningfulness of the structural analysis. Thus, it is not surprising that the resulting layout is fundamentally different from all other layouts (see Fig. 1).

If, however, influence is proportional to frequency and/or strategic positioning, the WordSpace layout can display its advantages. Especially the CRA procedure with betweenness centralities is able to capture a lot of structural information and to provide the a good starting point. Here similarity and influence analysis can reveal it's full potential resulting in nice visualizations revealing the most important topics in the collection and relations between them. For a more detailed view we provide in Fig. 2 a slightly bigger picture of the layout in Fig. 1. Furthermore, to verify the similarity analysis we give a detailed view of the words only with document labels removed. The more technical and mathematical topics appear in the lower part (network, graph, model, parameter, data, etc.) of the drawing while the objects of study and the interpretational terms (e.g. knowledge, communication, relation, firm, trust) are positioned in the upper part. Several interesting meaningful connections can be drawn of words that are located closely (e.g. university / research, study / student or the topic slice of data, with experimental, design, case, generator, population, mean, reliability, validity, etc.)

Visual layout features. Apart from the analytical, several visual adjustments have been made to create the images of Fig. 1. Although in all of variants presented the WordSpace layout strives to cover most of the space, it is likely to concentrate item labels in the center of the drawing. In Laplacian methods this is due to edge length minimization, in k-cluster drawing this happens to unimportant items with low eigenvector entries. To limit this effect, we relocated all items by scaling their distance to the origin with the 3rd or 4th root.
Another problematic issue is overlapping of item labels. Here we introduced an ordering such that more important items are drawn in front of less important items. Furthermore, importance corresponds to the darkness of the word color. This supports readability on white background. In addition we colored term labels in blue and document labels in red colors. Finally, word labels are drawn in lower case and document labels in upper case letters.
Not all information about documents can be displayed in an overview. However, we chose for each document a combination of the authors as label in order to quickly identify documents and researchers with descriptive keywords.
In order to further handle overlapping text a technique was adopted that was already successfully employed in other text visualization methods.[26] Each letter is surrounded with a small white border to intensify its contours. This small feature provides remarkable improvements on readability.

ACKNOWLEDGMENTS

Martin Hoefer acknowledges support by DFG Research Training Group 1042 "Explorative Analysis and Visualization of Large Information Spaces". Jürgen Lerner acknowledges support by DFG under grant Br 2158/1-2.

REFERENCES

1. V. Batagelj, A. Mrvar, and M. Zaversnik, "Network analysis of texts," in *Jezikovne tehnologije / Language Technologies*, T. E. . J. Gros, ed., pp. 143–148, Ljubljana, 2002.

2. U. Brandes and S. Corman, "Visual unrolling of network evolution and the analysis of dynamic discourse," *Information Visualization* **2**(1), pp. 40–50, 2003.
3. S. Corman, T. Kuhn, R. McPhee, and K. Dooley, "Studying compley discursive systems: Centering resonance analysis of communication," *Human Communication Research* **28**(2), pp. 157–206, 2002.
4. K. Andrews, W. Kienreich, V. Sabol, J. Becker, G. Droschl, F. Kappe, M. Granitzer, P. Auer, and K. Tochtermann, "The infosky visual explorer: exploiting hierarchical structure and document similarities," *Information Visualization* **1**(3-4), pp. 166–181, 2002.
5. J. Lamping, R. Rao, and P. Pirolli, "A focus+context technique based on hyperbolic geometry for visualizing large hierarchies," in *Proceedings of CHI 95: Human Factors in Computing Systems*, pp. 401–408, 1995.
6. K. Andrews, J. Wolte, and M. Pichler, "Information pyramids: A new approach to visualizing large hierarchies," in *Proceedings of IEEE Visualization '97*, pp. 49–52, 1997.
7. M. Chalmers and P. Chitson, "Bead: Explorations in information visualization," in *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 330–337, 1992.
8. M. Chalmers, "Using a landscape metaphor to represent a corpus of documents," in *Spatial Information Theory: A Theoretical Basis for GIS, International Conference COSIT '93, Proceedings*, I. C. A.U. Frank, ed., *LNCS*(713), pp. 377–390, 1993.
9. J. Wise, "The ecological approach to text visualization," *Journal of the American Society for Information Science* **50**(13), pp. 1224–1233, 1999.
10. J. Thomas, P. Cowley, O. Kuchar, L. Nowell, J. Thomson, and P. Wong, "Discovering knowledge through visual analysis," *Journal of Universal Computer Science* **7**(6), pp. 517–529, 2001.
11. M. Chalmers, "A linear iteration time layout algorithm for visualising high-dimensional data," in *Proceedings of IEEE Visualization '96*, pp. 127–132, 1996.
12. "IN-SPIRE visual document analysis." http://in-spire.pnl.gov/, 2004.
13. M. Berry, S. Dumais, and G. O'Brien, "Using linear algebra for intelligent information retrieval," *SIAM Review* **37**(4), pp. 573–595, 1995.
14. M. Berry, Z. Drmac, and E. Jessup, "Matrices, vector spaces and information retrieval," *SIAM Review* **41**(2), pp. 335–362, 1999.
15. L. Freeman, "A set of measures of centrality based on betweenness," *Sociometry* **40**, pp. 35–41, 1977.
16. M. Newman, "A measure of betweenness centrality based on random walks," 2003. http://arxiv.org/abs/cond-mat/0309045.
17. U. Brandes and D. Fleischer, "Centrality measures based on current flow," in *Proceedings of the 22nd Symposium on Theoretical Aspects of Computer Science (STACS)*, pp. 533–544, 2005.
18. G. Salton and C. Buckley, "Term-weighting approaches in automatic retrieval," *Information Processing & Management* **24**(5), pp. 513–523, 1988.
19. U. Brandes and J. Lerner, "Structural similarity in graphs," in *Proceedings of the 15th International Symposium on Algorithms and Computation (ISAAC '04)*, pp. 184–195, 2004.
20. C. Godsil and G. Royle, *Algebraic Graph Theory*, Springer, 2001.
21. Y. Azar, A. Fiat, A. Karlin, F. McSherry, and J. Saia, "Spectral analysis of data," in *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pp. 619–626, 2001.
22. G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*, Academic Press, 1990.
23. Y. Koren, "Graph drawing by subspace optimization," in *Proceedings of 6th Joint Eurographics - IEEE TCVG Symp. Visualization (VisSym '04)*, pp. 65–74, 2004.
24. P. Eades, "A heursitic for graph drawing," *Congressus Numerantium* **42**, pp. 149–160, 1984.
25. H. Zha and Z. Zhang, "Matrices with low-rank-plus-shift structure: Partial SVD and latent semantic indexing," *SIAM Journal of Matrix Analysis and Applications* **21**(2), pp. 522–536, 1999.
26. W. Payley, "Textarc: Showing word frequency and distribution in text." 2002 IEEE Symposium on Information Visualization (InfoVis 2002), 2002. available at http://textarc.org.

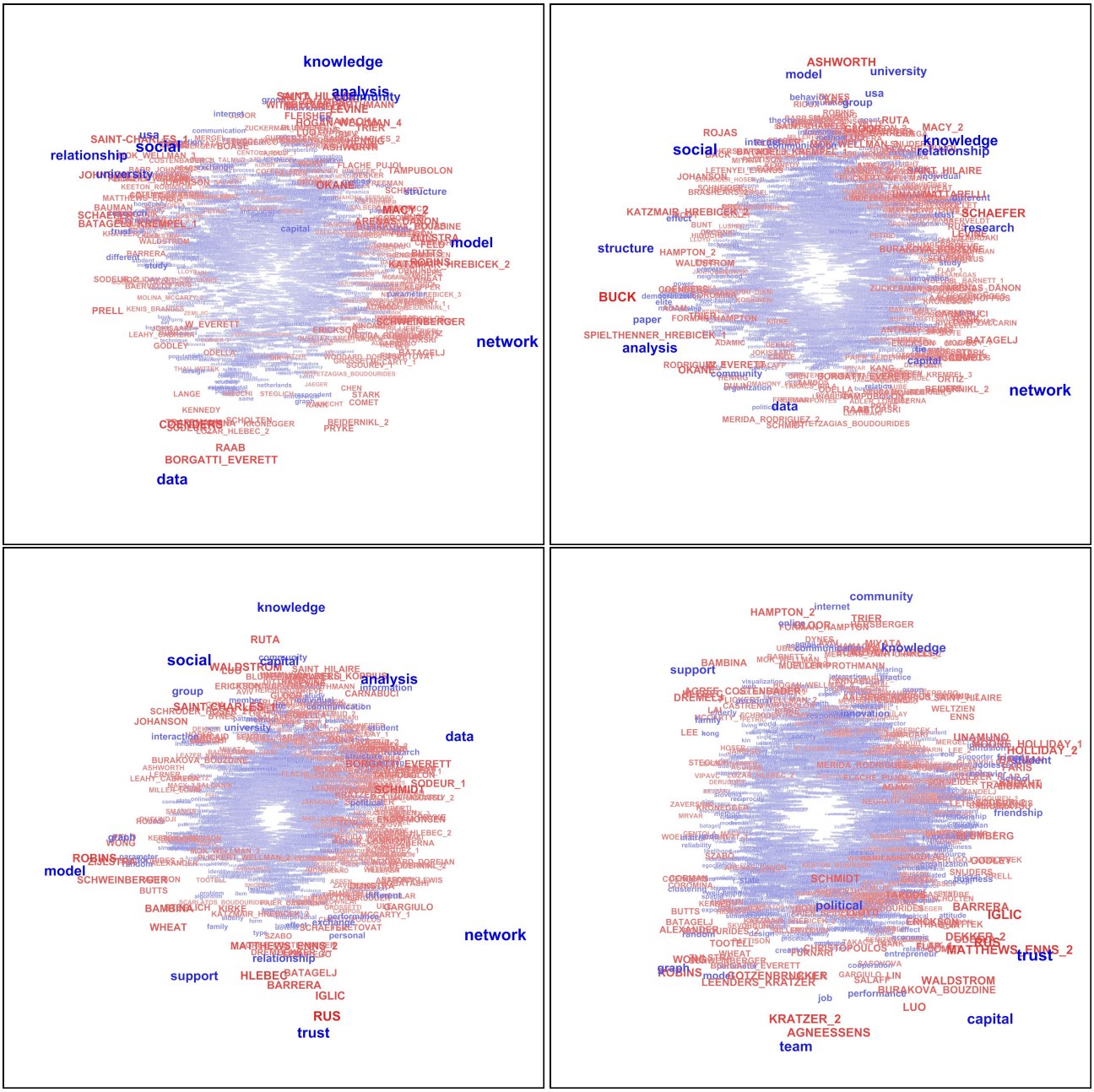

Figure 1. WordSpace Layout for Sunbelt 2004 abstracts with 8-cluster drawing method using shortest-path (upper left), current flow (upper right) betweenness, term-frequency (lower left) and TFIDF (lower right).

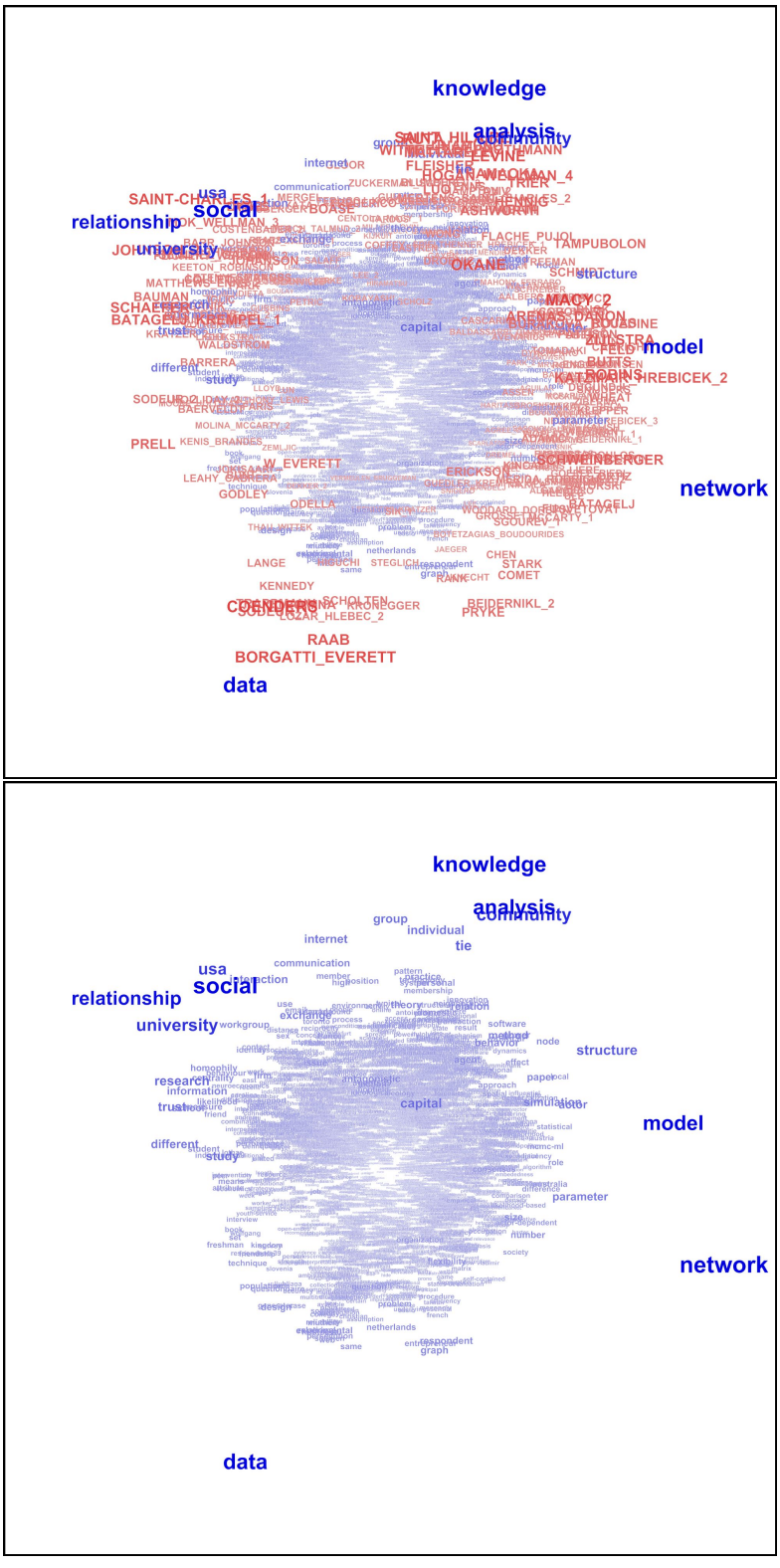

Figure 2. Layout from Fig. 1 with CRA and shortest-path betweenness (upper) and drawing of words only (lower).