UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação
ISSN 0103-2569

Fast Content-Based Visual Mapping for Interactive Exploration of Document Collections

Rosane Minghim
Fernando Vieira Paulovich
Alneu de Andrade Lopes

Nº 258

RELATÓRIOS TÉCNICOS

São Carlos – SP
Jun./2005

Fast Content-based Visual Mapping for Interactive Exploration of Document Collections

Rosane Minghim, Fernando Vieira Paulovich, and Alneu de Andrade Lopes

Instituto de Ciências Matemáticas e de Computação
CP 668, São Carlos 13560-970, São Paulo, Brazil
{rminghim,paulovic,alneu}@icmc.usp.br

Abstract. This paper presents a fast technique for map generation of document collections that, besides being able to group (and separate) documents by their contents, runs at very manageable computational costs, generating maps of pre-processed text in a matter of seconds. Based on multi-dimensional projection techniques and an algorithm for projection improvement, it results in a surface map that allows the user to identify a number of important relationships between documents and groups of documents that are reflected as visual attributes such as height, color, isolines as well as aural attributes (such as pitch). The map is interactive, allowing further exploration and narrowing of focus on a search task. The technique, named IDMAP (Interactive Document Map), is fully described in this paper. The results are bound to support a large number of applications that rely on retrieval and examination of document collections.

1 Introduction

In a quest for information on a particular subject in a large document data base, it is a fact that the exploration of the results obtained has become a tiresome task due to the amount of information retrieved by most search engines, and the lack of comprehensive structure in their responses. Most search engines, and other supporting tools for data interpretation, return the information about documents that, in spite of been progressively enhanced, does not allow the identification of higher level structure possibly revealed by the documents, such as new research areas and sub-areas.

In order to support the task of browsing through recovered sets of documents, there is a number of techniques for document ranking and for visualization and mining. Using such techniques, document maps can be generated to help users to identify the information contained in the documents, and the relationship between them. However, the level of information that users can extract from such maps still needs to be improved. In this context, new visualization supports must be sought to help the user to pursue useful information.

This paper presents a complete methodology for generation of document maps that improves the existing support to the identification of structure within

bodies of documents, as well as the identification of relationships between documents in a document collection. The methodology is based on multi-dimensional projection, resulting on maps that group similar documents effectively. As a result, it allows users to identify similarity, relevance, and sub-areas of activity based on document content. The display is a surface that can be explored using an interactive tool that allows users to examine individual document properties and neighborhoods.

The results are promising and are evaluated in terms of processing speed and effectiveness of representation as compared to other available text visualization techniques.

Section 2 presents a background review on text visualization techniques, highlighting the motivation for this work. Section 3 describes the full processing for document map generation as well as its justification. Section 4 discusses the results. Conclusions are drawn in Section 5.

2 Visualization of Texts

Text exploration and organization of text collections are tasks necessary in an increasing number of applications. Therefore new tools for better interpreting a set of texts without actually having to read them is paramount. The complexity of the information involved calls for more support from perceptually significant visualizations.

With the general goal of helping to organize search results, a number of different techniques for text sets visualization, resulting from Web and other repositories, have being deployed ([1], [17], [27], [2]). While these techniques make sense for large text bodies of varied information, our focus is to provide a map of documents that can be explored in several levels (from overview to individual text) to find structures inside more limited collections of texts. So we assume a pre-filtering task (such as a restricted search) that reduces the universe of targeted documents to a few hundred texts. This is the context in which the technique has been tested, although there is indication that many of the results can be extended to mapping general searches.

There are many techniques for text visualization that, in order to meet the above mentioned targets, search for a representation of the content of an individual text (eg. [20], [23]), of text sets (eg. [17], [5], [30]), or of themes approached in texts (eg. [13], [31], [32]).

Usually text processing tasks employ the vector space model [24] whereby texts are represented as vectors. In this vector space, each text is a vector with the dimensions represented by terms (n-grams). The coordinates are the weights of the terms based on their frequencies. Texts dimensions tend to be very high for collections of documents. Typically, those dimensions reach the thousands even for small to medium data bases.

The most common way to extract structure from a text collection is by applying some sort of dimensional reduction technique over that vector representation. This is the case of systems based on Multi-dimensional Scaling (MDS), Principal

Component Analysis (PCA) or Latent Semantic Indexing (LSI), that work with statistical measures for subspace reduction, and Self-Organizing Maps (SOM), that employ neural computation ([30], [31], [5], [16], [32]). Those techniques can be used to plot the original data in bidimensional (2D) space, when dimension is reduced to 2.

Although dimensionality reduction is a natural processing trend for texts, these types of techniques have high computational costs and low adaptability to incremental processing. Multidimensional reduction techniques also cause other difficulties, such as [14]: high information loss when applied directly to two dimensions (for display); reduction in input dimensions do not seem to affect greatly the outcome; and there is a inherent discretization problem associated with techniques such as SOM, by which individual documents in groups are not distinguishable. For the target of this work, dimension reduction poses and additional problem: when used to display the results in 2D, the mappings to subspaces may define groups of 'similar documents', but locally it is not possible to relate neighboring texts. In a previous work [18] we have applied LSI successfully for the generation of document maps, but at much higher cost than with the technique presented here.

Another recurring strategy to dealing with the organization of information from a text collection is document clustering ([7], [22]), many times employed in combination with dimensional reduction and SOM ([15], [30], [17]). They provide a way of relating documents with varying success rates. When clustering techniques are applied, the intra-cluster relations are also not given as a result. However, they are very useful to provide general overviews of large collections, although they usually have to be interpreted by users with certain level of expertise.

There are approaches that completely avoid the problem of high dimensionality by simply ordering the most used terms in the text and employing the first N terms [23]. These strategies work well for single text representation and for association of a limited number of texts, and even for some degree of clustering. However, it also lacks in association between different documents and definition of levels of similarity. Other approaches (such as the one by Carey and others [7]) combine a number of different strategies to allow various views of the same document set, potentially improving focusing and analysis tasks.

In general, the methods discussed above lack the ability to determine levels of associations between texts. Others, that provide grouping (that is, more significant information), are computationally expensive. Most do well with displaying overall views of document collections but fail to support focussing towards smaller groups of similar texts. We refer to the work of Katy Borner and others [6] for a detailed description of the available techniques for text mapping and their challenges. In many aspects the technique presented here is a complement to those, providing means of rapidly examining in-context groups of documents with good overall view as well. Its central points are the use of faster distance-based projection techniques followed by a technique to recover some of the information loss in the process of projection.

The visual representation adopted here is the landscape-type of display, which is very useful due to its ability to reveal information without resorting to highly attentive perceptual processes, allowing interpretation even by users with little expertise in the field. It has been the choice of many useful presentations of texts before ([30], [31], [8], [9]). Additionally, surfaces are highly interactive and familiar to most users. The surface representation of our technique is enriched by mapping further significant information (such as degree of similarity) to visual attributes (such as lines, colors and height) and aural attributes (such as pitch and timbre). The final map can be explored by the users interested in having an overview of a set of texts, locating important texts in the corpora, or finding useful associations between texts.

The next section presents the complete mapping process from pre-processing to projection and attribute mapping procedures.

3 Projection Techniques for Text Visualization

A previous work [29] has shown the advantages of projection techniques to obtain useful views of multi-dimensional data sets based on distance metrics. Additionally, those techniques perform well in terms of processing speed and lend themselves to landscape plots. This work set off to decide whether those techniques could be as successfully applied to representation of text sets.

Different from other techniques that can be used to map data into 2D or 3D, such as dimensional reduction and clustering, the goal of distance-based projection techniques - eg. Fastmap [12] and Nearest Neighbor Projection (NNP) [29] - is to place a set of points defined in multi-dimensional space in another space such that the relative distances between points are preserved as much as possible. The degree to which that distance cannot be preserved is called the error of the projection. For projections into a bidimensional space (plane), this problem can be stated as:

Let X be a set of points in $\mathbb{R}^n$ and $d : \mathbb{R}^n \rightarrow \mathbb{R}$ be a criterion of proximity between points in $\mathbb{R}^n$. We wish to identify a set of points P in $\mathbb{R}^2$ such that if $\alpha : X \rightarrow P$ is a bijective relation and $d_2 : \mathbb{R}^2 \rightarrow \mathbb{R}$ is a proximity criterion in $\mathbb{R}^2$, then $|d(x_i, x_j) - d_2(\alpha(x_i), \alpha(x_j))|$ is as close to zero as possible $\forall x_i, x_j \in X$. In this paper we call the set P a projection.

In order to create meaningful projections, the definition of a proper proximity criterion between the points of X is very important. Here, each document is represented as a vector, so that it is possible to calculate algebraically the distance (proximity) between texts.

The complete set of steps taken to build a map based on projection in this work are:

1. text corpus pre-processing to build its representation in vector space;
2. projection to two-dimensional space using a fast algorithm, followed by an improvement strategy (the Force Scheme [29]);
3. hierarchical clustering of the projected data for subgroup identification.

The next sections describe each step of the process in turn.

3.1 Text pre-processing and pre-reduction

In order to generate the vector representation of the text set in this work, the original texts (composed by title, authors, abstract and references of articles) were submitted to the following procedure:

1. Stopwords were eliminated from all texts;
2. Stemming was applied to extract word radicals using Porter's algorithm [21].
3. A frequency count was performed applying Luhn's cut [19] so that terms appearing less then 5 times were ignored.
4. Bi-grams were formed from the remaining words in the texts, that is, we considered as terms the occurrence in sequence of a pair of words. The software employed in this step was *Ngram Statistic Package (NSP)* [3].
5. A process to weight terms according to their frequency was carried out; in our case the weight was computed as the *term-frequency inverse document-frequency (tfidf)* [25].

The result of that process is a table of documents x terms in which each term is weighted according to the tfidf. Each line of the matrix (a document) is a vector, each final bi-gram is a dimension, and the tfidfs are the coordinates.

For many text processing activities, the number of dimensions in the resulting table is still too large (usually it reaches few thousands). Since our methodology is based on the idea of preserving the original space distances into the projected space, and the distance between points become unstable with high dimensionality [4], a pre-reduction of attributes (terms) must be performed before the projection takes place

One effective way to deal with this pre-reduction is to define new 'groups of terms' by attribute clustering to obtain a new reduced set of attributes. To do that, k-means clustering can be successfully employed [10] taking terms as vectors and documents as attributes (that is, provisionally transposing the representation). Due to the charge of that procedure in the overall processing time, we have devised a change in the algorithm (see Figure 1) with little influence in the outcome of the projections.

The difference to the original k-means lies in the fact that the centroid is recomputed each time a point is assigned to it. After centroid selection, the matrix terms x documents is recomposed by combining the values of the attributes in the same group to form one single attribute.

The above pre-reduction results on a new matrix $T_{n \times l}$ of documents with the same n documents but l 'clustered' terms. The coordinate for the new terms is a combination of the *tfidf* (term frequency inverse document frequency) of the various terms in the cluster it represents [10].

3.2 Projection techniques

In order to execute a projection based on the matrix T, a metric is necessary to establish a distance criterion between two different texts (d in the problem statement given previously in this text). A similarity measure that usually performs well for document processing is the cosine metric, given by:

Inputs:
$A = \{a_1, \ldots, a_m\}$ (attributes to be grouped)
l (number of groups)
N (number of iterations)
Output:
$C = \{c_1, \ldots, c_l\}$ (centroids)
Procedure:
Set C to initial value (e.g. random selection of A).
For x from 1 to N
For each $a_i \in A$
Find the nearest centroid $c_j \in C$ of a_i
Recompute the centroid c_j according to a_i

Fig. 1. Modified k-means feature clustering.

$$scos(t_i, t_j) = \frac{\sum_{k=1}^{l}(t_{ik} \cdot t_{jk})}{\sqrt{\sum_{k=1}^{l}(t_{ik}^2) \cdot \sum_{k=1}^{l}(t_{jk}^2)}} \tag{1}$$

where:
t_i and t_j are two lines of the matrix $T_{n \times l}$ of documents, that is, the vector representations of two different documents.

From that, a distance metric can be obtained [12], given by:

$$d(t_i, t_j) = \sqrt{2 * (1 - scos(t_i, t_j))} \tag{2}$$

Two distance based projection techniques were used in this work to generate the projection P of the set T of points. The first was Fastmap and the second was the NNP, with similar results. Fastmap [12] is a well know technique based on the idea of hyperplane projection. NNP [29] projects points defined in the original space onto 2D plane by trying to find a position that reproduces the original distance between each point to its 2 closest points. Details of both techniques can be found in the references provided.

In the context of multi-dimensional visualization, the original Fastmap tends to produce representations too compact while NNP tends to scramble neighboring points, impairing the effectiveness of the final result. A force-based projection improvement technique (see [29]) was used to enhance point placement, recovering part of the information lost during the projecting process. What this force scheme does is iteratively approach points projected too far and repel points that were projected closer than they should have. This algorithm is quite fast as applied in the context presented here.

The final result of the projection is a set of points P in $\mathbb{R}^2$, each representing one document, so they can be plotted using their projected coordinates.

Visualizing the projected documents as points on a plane tends to compromise the interpretation and exploration, even when the points are represented

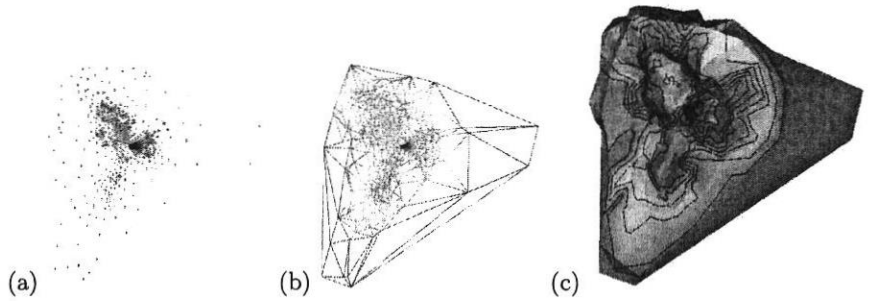

Fig. 2. Generation of a surface model from text data. (a)'glyphed' view of the projected points (b) 2D Delaunay Triangulation (c) Isolines and height map for subgroup determination.

by ghyphs such as a sphere (see Figure 2(a)). In order to produce a better landscape view from the projection, we perform a Delaunay triangulation [11] over the vertices (see Figure 2(b)).

The triangulation is useful for many purposes besides improving the perception of neighborhood. It lends itself to other mappings that can be combined to highlight important information to the user. The subject of attribute mapping is discussed in the next section.

3.3 Attribute mapping

The projection gives a placement of documents on a plane, where the document positions are based on a similarity measure of documents. The triangulation offers a mesh of these data, whereby the neighbors of a particular point (vertices in incident edges) indicate the closest documents according to the projections.

Classically, visualization techniques can generate a number of graphical displays from this set-up, particularly if scalar (or even vector) data are assigned as attributes to the vertices of the mesh. In the display, colors or heights can help locate highlighted attribute information. Paper relevance, number of citations, year of publication, are all valid attributes in that context.

Another possibility we have considered for attribute assignment is to use it to reflect sub-areas of interest inside the main subject areas of the document search. To do that, we have performed *Hierarchical Clustering (HC)* of the projected data. HC defines groups of elements progressively, using divisive or agglomerative approaches. The approach used here adds elements in pairs (elements are first points then clusters), producing a tree (dendogram) with various levels of clustering. The decision of similarity between clusters was based on the single-link method [28]. As well as the similarity within a group, the depth of the dendogram formed is meaningful information as well, in the sense that it represents a level of grouping with other points.

Therefore by mapping the depth of the HC to each projected point (as a scalar stored in the triangulation), new visual mappings can be obtained of levels of similarity among documents.

Figure 2(c) shows the mapping of that information (clustering depth) to color. In that case, the points colored in dark blue (or green) can be seen as the focuses of the various clusters of documents, that is, the documents placed at the leaves of the dendogram produced by the hierarchical clustering process. As the color changes from blue to red (in the rainbow color scale) new levels of grouping are achieved, until clusters merged closer to the root of the tree are viewed in the red regions. In our initial maps, that same information is redundantly mapped to height. Thus, the leaves of the clustering tree are also at the hill peaks, and the roots are the valleys of the landscape. In addition we have also traced level curves (isolines) corresponding to various clustering levels, in order to facilitate the visualization of the sub-groups formed, thus completing the map. From this figure it is possible to see that the curves help identify the borders of groups of documents.

Many of the features that a map is capable of representing are more useful under interactive exploration. The possibilities of interaction with the map are presented in the following section.

3.4 Interacting with surface maps of documents

The *spider cursor* is a tool for interaction with visualizations using sound [26]. The tool allows exploration of a data set represented by a triangulation by showing a cursor (called *spider*) on top of the triangulation as the user moves the mouse over the surface. From a central point (located by the cursor) the neighbors in the triangulation are shown with line segments (looking like spider legs). The value stored in the vertex pointed by the cursor is mapped to pitch of a pre-selected instrument. Another document property, represented by a character string, can be shown on a field on top of the presentation window. Section 4 shows some screen shots of maps under exploration by the spider cursor (see Figure 5). The tool complements the static attribute mappings for the interpretation of landscape displays.

The sound mapping is very useful, amongst other things, to resolve ambiguities in the visual mapping. Thus, two different documents visually undistinguishable in terms of their color or height can most times be told apart by the sound they produce when pointed at.

The techniques presented here (jointly called IDMAP - *Interactive Document Map*) resulted in a very useful interactive map for exploration of text collections (and other data). The next section presents some results obtained with the processing of various corpora using this methodology.

4 Results

Using the IDMAP method presented here we have initially processed three corpora (one on Case-Based-Reasoning (CBR), another on Inductive Logic Pro-

gramming (ILP), and a third on Information Retrieval (IR)) comprehending a collection of 574 documents. The CBR and ILP corpora were manually extracted from Lecture Notes on those subjects and the IR corpus resulted from a web search on the subject. All the pre-processing of the text to extract the contents and normalize the references was done by student members of our teams.

Reviewing the process, from the pre-processing step we have as result a matrix of documents x terms. That matrix was 574 documents x 6082 attributes for the CBR+ILP+IR corpus using a Luhn's cut off of 5. After that, there is a further reduction using (modified) K-means feature clustering. Both the original and the modified K-means were used, and the results are shown here for comparison.

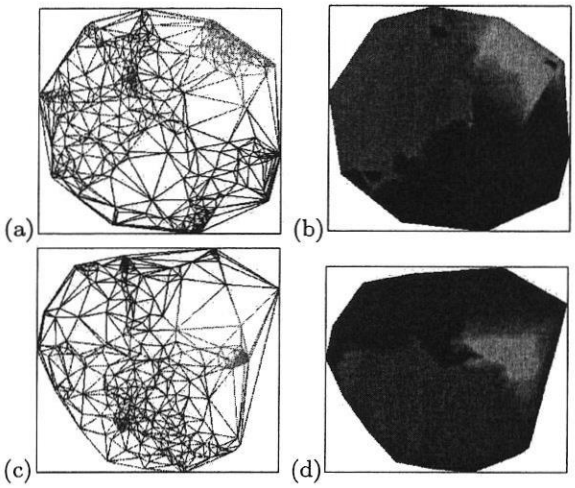

Fig. 3. Projections of the CBR+ILP+IR corpus using NNP. Color is pseudo-class. Red is CBR. Green is ILP. Blue is IR. (a) 2D Delaunay triangulation of the projection with k-means feature clustering (b) Resulting surface view of (a) (c)2D Delaunay triangulation of the projection with modified k-means feature clustering (d) Surface view of (c)

That resulted in a document matrix, for the triple corpus, of 574 documents x 13 attributes. These documents were then mapped using NNP projection. Figure 3 shows the outputs of this process, where color represents at this point the original classification of the documentes as CBR, ILP, or IR. This is a pseudo-classification, since the source of the paper was the only criterion to determine its class. It is, however, a good basis for evaluating the results.

As it can be observed in those pictures, the projections come a long way towards separating the main classes of documents. They occupy specific 'portions' of the map, apart from a small proportion of 'outliers' clearly distinguishable in the surface views of the projections. The fuzzy boundaries as well as the outliers are expected, particularly because the classes were inferred from their sources,

and the three areas of knowledge have many concepts (and certainly expressions and terms) in common.

To verify the ability to split document collections by content, we have processed and added a new set of documents, containing 101 documents on the general subject of sonification [1]. The corpus was recovered from a web search and processed the same way as the others. Figure 4 shows the result of processing the new corpus (SON) together with the other three test corpora. It can be seen from that projection that the technique still separates the new corpus from the three others. As expected, there is an extra number of outliers due to the addition of a new class.

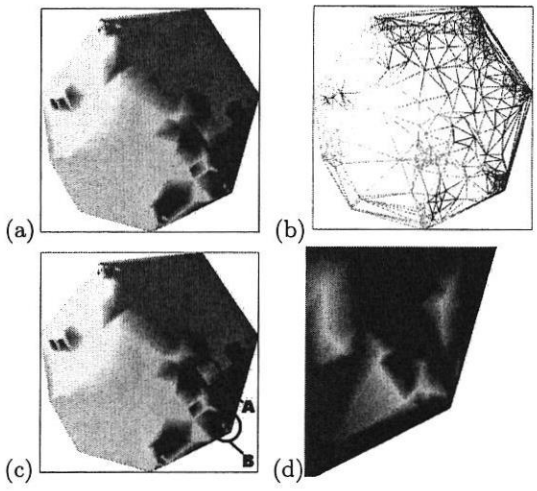

Fig. 4. Projection of the CBR+ILP+IR+SON corpus, using FASTMAP with modified k-means feature clustering. Color is pseudo-class. Red is CBR. Light blue is IR. Dark blue is SON. Yellow is ILP. (a) Surface view (b) 2D Delaunay triangulation (c) Groups of papers intentionally added (d) An amplified part of (c).

In order to test the capability of content based point placement, another test was performed - seven papers were intentionally choosen and added to the group of papers. Five of them belong to the general class of sonification. These articles had at least two common authors and represent and evolution of the same sonification system over the years. The other two papers again have at least two of the authors repeated, but the main subject of the paper was not sonification. One of them mentioned sonification incidentally, the other didn't. Figure 4 shows the map for that test.

In Figure 4 (c) is possible to note that the new papers belonging to the sonification area are mapped within the general sonification pseudo-class (circle B). The paper that mentioned sonification incidentally, is also mapped within

[1] *Sonification* is defined as the display of information through sound.

this pseudo-class, indicating the importance of author's names (and most likely some self-reference too) in the context. The paper that is not related to the sonification area (circle A) is placed between two groups of different pseudo-classes, indicating that it is not strongly related to any of these groups. In addition, the papers that deal with the same sonification system are grouped together since they are quite related. Figure 4 (d) shows an amplified part of figure (c) which represents the circle B.

The final document map is obtained, as detailed in Section 3.3, by running hierarchical cluster over the projection. Figure 5 shows the final maps of four-corpora recovered from an Internet repository[2]. It comprehends 1624 files in the ISI format on the subjects of Bibliographic Coupling (BC - in red), Cocitation Analysis (SC - in blue), Milgrams (MB - in green) and Information Visualization (IV - in orange). It can be seen in those pictures that the IV corpus, which dominates the plot due to the number of files it comprises (1236 files), possesses five 'hill peaks', while other 'hill peaks' on the map also appear, one for each of the other classes (once BC and SC are correlated areas, there is only one 'peak' for both of them).

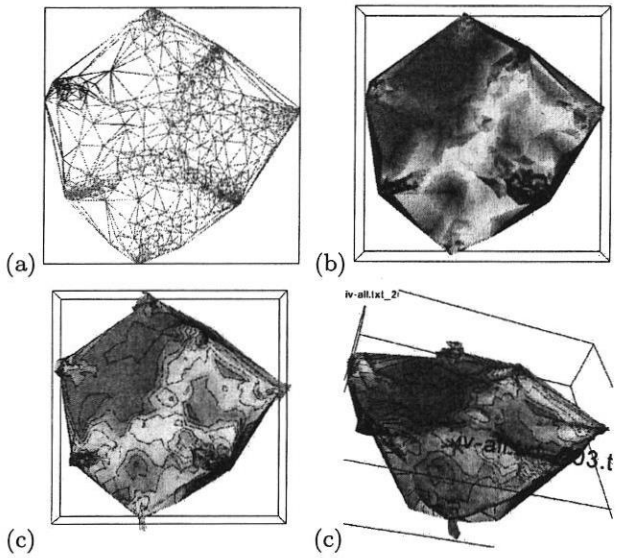

Fig. 5. Final document collection map after projection by FASTMAP with modified k-means. Color and height are cluster depth (level of grouping). Isolines limit sub-areas. Hierarchical clustering is single link. (a) Map of BC+IV+MB+SC corpus (b) Map of BC+IV+MB+SC in perspective (c) Map of BC+IV+MB+SC with level curves. (d) Spider cursor over the map of BC+IV+MB+SC corpus.

[2] ella.slis.indiana.edu/~ katy/outgoing/hitcite/{bc,sc,mb,iv}.txt

The displays were generated using the visualization toolkit (http://www.vtk.org), of which the spider cursor exploration tool is an extension.

Preliminary results of exploration of these maps indicate that the subgroups formed in the visualizations match the general idea of close association between papers, with few exceptions. Closer examination (both theoretical and practical) of that issue is being pursued.

The whole process, from the creation of the original term matrix (after bi-gram determination) to attribute mapping, takes 24.109 seconds for the CBR+ILP+IR+SON corpus using k-means feature clustering and 13.36 seconds using modified k-means feature clustering; 142.187 seconds for the BC+IV+MB+SC corpus using k-means feature clustering and 106.596 seconds using modified k-means feature clustering. These times were reached on a AMD Athlon XP 2400 computer, 2 GHz, and 512 MB of RAM. Surface display itself (including isolines calculation) is instantaneous. Interaction is done by the spider cursor in real time.

As far as algorithm complexity goes, feature clustering is $O(n)$, FASTMAP is $O(log(n))$, NNP is $O(n^2)$ (or $O(log(n!)$ for data access with fast nearest neighbor query). Force is $O(n^2)$. With IDMAP the algorithms with higher complexity order are run on lower dimensionalities, which improves computational costs considerably.

The above maps generated by IDMAP were compared with the ones we have produced in a preliminary work of ours [18]. That work created maps by means of dimension reduction using LSI. We have used the package SciLab (http://scilabsoft.inria.fr/) to process a dimension reduction of the terms $\times$ document matrix to 2, plotting the result as 2D coordinates. In terms of quality, LSI maps tend to separate well up to three classes of texts, with slightly less outliers. However, when the class SON was added to the previous 3-subject corpus, there was a mixture between SON and IR documents. Comparatively, IDMAP has separated well the 4-subjects corpora with no difficulty. Also, computational times for LSI based maps ran in the order of minutes due to the need for processing the data without attribute reduction to obtain good results. For instance, for the corpus CBR+ILP+IR+SON it took 159.203 seconds to generate the map; for the corpus BC+IV+MB+SC the use of computer memory was overwhelming and the LSI process did not complete. IDMAP, on the orther hand, completed in very reasonable time, as highlighted above, and used very little memory compared to LSI maps.

5 Conclusions

In terms of usability for exploration of texts, IDMAP compares well with other techniques. It complements those in the sense that it can generate maps that help explore structure and relationships in document collections inter and intra-groups and allow exploration of those collections in various levels (from overview to individual browsing). It is fast enough to deal with reasonably large corpora, comparing very well against the alternative use of dimension reduction, and generates various levels of information. In itself it presents a novel, unique form of

document map generation, to support information gathering prior to individual text examination.

A very important observation on the technique proposed is that, although a few outliers appear after projection, the approach is still capable of separating groups of documents very well, at a rate close to the best classification techniques.

In a near future we plan to study the inter-relationships within the map to evaluate and model the effects of the various processes carried out against various other methods of clustering and document organization. Extensions of the mappings are planned to reflect higher order information and stronger semantic relationships amongst the documents.

A software system itself is to be developed and made available as a Web system for general use. All the pre-processing tasks that were performed manually in places are possible to made automatic, and this is in our plans.

The mapping process can be applied, from the distance matrix, to any data set that can be expressed that way, so the system will encompass other forms of data visualization and exploration apart from text.

We are working on other types of similarity metrics between texts to extend the flexibility and usability of IDMAP. Additionally, algthough scalability was not an initial focus here there is indication that the approach could be adapted to handle much larger text collections. That subject is under study.

References

1. O. Alonso and R. Baeza-Yates. Alternative implementation techniques for web text visualization. In *Proc. of the First Latin American Web Congress (LA-WEB 2003)*, pages 202–203, Santiago, Chile, November 2003. IEEE Computer Society, IEEE Press.
2. M. R. Baeza-Yates. Visualizing large answers in text databases. In *Int. Workshop on Adv. User Interfaces (AVI'96)*, pages 101–107. ACM Press, 1996.
3. S. Banerjee and T. Pedersen. The design, implementation, and use of the Ngram Statistic Package. In *Proc.of the 4th Intl. Conf. on Intel. Text Proc. and Comput. Ling.*, Mexico City, February 2003.
4. K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is nearest neighbor meaningful? In *Lecture Notes in Computer Science*, volume 1540, pages 217–235, 1999.
5. A. Booker, M. Condliff, M. Greaves, F. B. Holt, A. K., D. J. Pierce, S. Poteet, and Y. J. Wu. Visualizing text data sets. *Computing in Science and Engg.*, 1(4):26–35, 1999.
6. K. Borner, C. Chen, and K. Boyack. Visualizing knowledge domains. *Ann. Rev. of Inf. Sci. & Techn.*, 37:1–51, 2003.
7. M. Carey, D. C. Heesch, and S. M. Ruger. A visualization tool for document searching and browsing. In *Proc. of Intl Conf. on Distri. Multimedia Sys.*, 2003.
8. M. Chalmers. Using a landscape methaphor to represent a corpus of documents. In Andrew U. Frank and Irene Campari, editors, *Spatial Information Theory: A Theoretical Basis for GIS*, volume 716 of *Lecture Notes in Computer Science*, pages 377–390. Springer, 1993.

9. M. Chalmers and P. Chitson. Bead: Explorations in information visualization. In *Proc. ACM SIGIR*, pages 330–337. ACM Press, 1992.
10. I. Dhillon, S. Mallela, and R. Kumar. Enhanced word clustering for hierarchical text classification. In *Proc. of KDD-02*, pages 191–200, Edmonton, CA, 2002. ACM Press, New York, US.
11. H. Edelsbrunner. *Geometry and Topology for Mesh Generation.* Cambridge Monographs on Applied and Computational Mathematics. Cambridge, 2001.
12. C. Faloutsos and K. Lin. Fastmap: A fast algorithm for indexing, datamining and visualization of traditional and multimedia databases. In *ACM SIGMOD Intl Conf. on Manag. of Data*, pages 163–174, San Jose-CA, USA, 1995. ACM Press: New York.
13. S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Trans. Visual. and Comp. Graphics*, 8(1):9–20, Jan-Mar 2002.
14. S. Huang, M. Ward, and E. Rundensteiner. Exploration of dimensionality reduction for text visualization. Technical report, 2003.
15. S. Iritano and M. Ruffolo. Managing the knowledge contained in electronic documents: a clustering method for text mining. In *12th DEXA Workshop*, pages 454–458. IEEE CS Press, 2001.
16. S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Websom - self-organizing maps of document collections. *Neurocomputing*, 1(1-3):110–117, 1998.
17. A. Leuski and J. Allan. Lighthouse: Showing the way to relevant information. In *InfoVus*, pages 125–130. IEEE CS Press, 2000.
18. A. A. Lopes, R. Minghim, and V. Melo. Creating interactive document maps through dimensionality reduction and visualization techniques. Technical report, São Carlos - SP, Brazil, 2005.
19. H. P. Luhn. The automatic creation of literature abstracts. *IBM J. of Res. and Develop.*, 2:159–165, 1958.
20. N. E. Miller, P. C. Wong, M. Brewster, and H. Foote. Topic islands - a wavelet-based text visualization system. In *Proc. of Visualization '98*, pages 189–196, Research Triangle Park, North Carolina, United States, 1998. IEEE CS Press.
21. M. F. Porter. An algorithm for suffix striping. *Program*, 14(3):130–137, 1980.
22. M. Rasmussen and G. Karypis. gcluto - an interactive clustering, visualization, and analysis system. Technical report, 2004.
23. R. M. Rohrer, D. S. Ebert, and J. L. Sibert. The shape of shakespeare: Visualizing text using implicit surfaces. In *IEEE InfoVis'98*, pages 121–129. IEEE Press, 1998.
24. G. Salton. Developments in automatic text retrieval. *Science*, (253):974–980, 1991.
25. G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988.
26. V. Salvador and R. Minghim. An interaction model for distributed visualization using sound. In *Proc. of Sibgrapi 2003*, pages 132–139. IEEE Computer Society Press, 2003.
27. M. M. Sebrechts, J. Cugini, S. J. Laskowski, J. Vasilakis, and M. S. Miller. Visualization of search results: A comparative evaluation of text, 2d, and 3d interfaces. In *22nd ACM SIGIR*, pages 3–10. ACM Press, 1999.
28. P. H. A. Sneath and R. R. Sokal. *Numerical Taxonomy: The Principles and Practice of Numerical Classification.* W.H. Freeman, San Francisco, 1973.
29. E. Tejada, R. Minghim, and L. G. Nonato. On improved projection techniques to support visual exploration of multi-dimensional data sets. *Information Visualization J.*, 2(4):218–231, 2003.

30. E. Weippl. Visualizing content based relations in texts. In *Proc. 2nd Australasian Conf. on User Interface*, pages 34–41. IEEE CS Press, 2001.
31. J. A. Wise. The ecological approach to text visualization. *J. of the American Soc. for Inf. Sci.*, 50(13):1224–1233, November 1999.
32. J. A. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow. Visualizing the non-visual: spatial analysis and interaction with information for text documents. In *Readings in information visualization: using vision to think*, pages 442–450, San Francisco, CA - USA, 1995. Morgan Kaufmann Publishers Inc.