

Visual Mining of Text Collections

Rosane Minghim¹ and Haim Levkowitz²

¹ICMC - Instituto de Ciências Matemáticas e de Computação
University of São Paulo
Brazil

²Institute for Visualization and Perception Research
University of Massachusetts Lowell
USA

Abstract

What happens if you have to examine and reach conclusions from a considerable number of textual documents? If you are faced with this task or with developing tools for completing this task, this tutorial is for you.

Examining text is crucial for many different types of applications. Even applications that rely on additional types of data (such as images, signals, simulations) usually have complementary or alternative text based output. The challenge of interpreting content and extracting useful information from a document collection is the target of efforts in various areas of computer science. Fields such as Text Mining try to extract knowledge automatically or semi-automatically from collected textual information; however, due to the multi-dimensional characteristics of text it is paramount to couple these algorithms with meaningful visual representations in order to improve performance and allow the discovery of relevant information within a text data set.

Since it is not feasible to go through the entire documents' content in detail due to data sizes and time constraints, Visual Text Mining (VTM) — the combination of Text Mining and Visualization — is focused on developing tools to help users extract meaning from text collections without extensive reading.

In this tutorial we introduce the necessary background and the graphical techniques involved in Visual Text Mining of document collections.

Contents

- 1 Overview, motivation, goals
- 2 Two test cases: Visual maps of flash news and scientific papers
 - 2.1 Visual mapping of news flashes
 - 2.2 Visual mapping of scientific papers
- 3 Basic Concepts
 - 3.1 Text processing and information retrieval
 - 3.2 Data and text mining
 - 3.3 Projection techniques
 - 3.4 Visual representations: graphs, surfaces, volumes, triangulations
- 4 From Visualization to Visual Text Mining
 - 4.1 Visualization techniques for multidimensional data
 - 4.2 Visualization techniques and systems for handling document collections
 - 4.3 Visual text mining
- 5 Projection Based Visualization and its Application to Visual Text Mining
 - 5.1 Projection techniques and point placement strategies
 - 5.2 Mapping text collections via projections and point placement
 - 5.3 Topic extraction and visualization
 - 5.4 Further Examples
- 6 Conclusions, Current Challenges, Future Trends
- 7 Acknowledgments

Appendices

- A Technical Report: VISUAL MAPPING OF TEXT COLLECTIONS USING AN APPROXIMATION OF THE KOLMOGOROV COMPLEXITY
 - A.1 Introduction
 - A.2 Previous Work
 - A.3 Projection techniques for text visualization
 - A.4 Kolmogorov Complexity as a means to define distance between texts
 - A.5 Results
 - A.6 Conclusions
- B Technical Report: CONTENT-BASED DOCUMENT MAPS USING FAST PROJECTIONS AND TOPIC DETECTION
 - B.1 Multidimensional Projections for Mapping Collections of Documents
 - B.2 Exploring Content-based Document Maps
 - B.3 Further Remarks

References

1. Overview, motivation, goals

Text collections are generally considered data sets with a high number of dimensions, where a dimension is a term or expression (an n-gram) of importance within the domain of the document collection. In a conventional vector representation of a document data set, the number of dimensions for a few hundred documents of moderate size can reach a few thousand.

Regardless of these difficulties and their implications for the analysis of text collections, there is a growing number of applications that can benefit from tools to effectively support analysis of document sets and from that analysis, allow the user to reach conclusions and to make decisions. The range of potential applications vary from health studies and diagnosis based on medical records to investigation of unlawful activities. The nature of the textual source is also quite varied. Documents can be snippets from web searches, RSS feeds of various kinds, scientific papers, reports, newspapers articles, automatically generated health test reports, patents, and so on. Every day new applications start to rely on text solely or combined with other data sources (such as table data and images). There is, therefore a compelling need for tools that combine user driven and automatically extracted displays to support the analysis of text collections and relationships amongst texts.

Finding ways to support analysts in quickly and meaningfully extracting meaning from text collections is also a strategic issue for the area of Visual Analytics, defined as "the science of analytical reasoning facilitated by interactive visual interfaces" (from <http://nvac.pnl.gov/>). The fields of Text Mining and Visual Text Mining aim at combining visualization and mining approaches to achieve solutions for the exploration of text collections.

Handling high-dimensional data poses many problems to researchers and data analysts in general. Visualization techniques seek to bridge the gap between users' visual perception and reasoning capabilities and analytical techniques. Nonetheless, finding intuitive ways to visualize large high-dimensional data sets is a difficult problem. Traditional multidimensional visualizations, such as scatter plot matrices, parallel coordinate plots or pixel-based techniques [OL03] operate by mapping each data attribute into a corresponding visual axis or other representation. As such, they can only handle well a very limited number of attributes and, therefore, are not directly applicable to complex data consisting of many attributes (dimensions) [MPL06].

A widely explored alternative to handle this type of data is to reduce dimensionality prior to visualization, e.g., by projecting the high-dimensional data points into a lower dimensional space (2D, 3D) that is more amenable to user interpretation. Various techniques exist to achieve such a mapping, mostly based on dimensionality reduction, dimensionality clustering, or point placement strategies. The resulting

projection can be displayed through a suitable visual representation — e.g., points placed on a plane, graphs, surfaces or volumes — that can be navigated and explored by a data analyst. Projections may be created based on different criteria, but typically they strive to preserve distance relationships amongst data points as defined in the original space. Information loss is inevitable and the extent to which this distance-preserving goal is achieved depends on the precision of the projection.

Figure 1 depicts the overall process in generating interactive visual representations of document data sets via projections (or, in fact, other point placement strategies too). The user can feed the system with (1) a document collection, (2) a structured data table, or (3) a distance matrix. In the first case, documents are either converted into a vector representation or compared using a similarity measure that handles text against text. From vector representations, distance calculations can also be performed. From distance relationships the 2D projection is generated and the user can interact with its visual representation to gain insight to the data set.

This tutorial treats the problem of mapping and exploring, through visual representations, textual data sets and results of information extraction from textual data sets of various different natures. We describe the basic principles for generating such visualizations and make an effort to convey the benefits of achieving a solid graphical view of automatically extracted relationships. The problem of facilitating text analysis is by far an open issue. We hope to identify, suggest and explain the current technologies available and the challenges ahead for professionals dedicated to both using and developing techniques for that purpose. In this tutorial, we offer an overview of the existing techniques underlying techniques to allow the construction of visual document maps from unstructured data sets, that is, without relying on previously extracted meta data. The resulting map should offer insight into the contents of the document data set to support examination and relief the user from extensive reading otherwise necessary.

We begin in the next Section by illustrating the creation and exploration of visual maps of example text collections of very different natures in two quite diverse areas of application, by use of projection techniques.

Following that we describe, in Section 3, some of the most important basic concepts and techniques involved in this area.

Section 4 reviews the techniques published in the literature to achieve some of the goals stated here.

Projection based visualizations and their use for visualization of text collections are detailed in Section 5. Finally, we discuss other underlying mining issues that go together with text analysis, such as Topic Extraction.

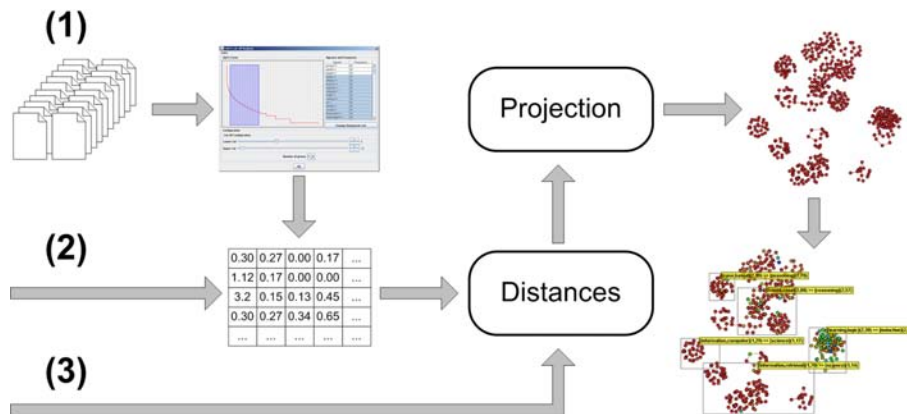


Figure 1: Generating projection maps of multidimensional data sets.

2. Two test cases: Visual maps of flash news and scientific papers

Visualizations of text documents and their meta-data can take many forms, whether or not they are coupled with data mining techniques. Since our focus is on non-structured data, here we illustrate the mapping based on content of two document collections bearing very different characteristics. The goal would be, through an analysis process involving such maps, to be able to extract and map features of the data set in a way that can help locate patterns. In each application, however, what needs to be located is different. In the first case, the mapping of news flashes, the target is to have an overview of the main events that have been reported on Web sites of some relevant news agencies. The second case illustrated here, the mapping of scientific papers, highlights the possibilities of identifying general subjects as well as specific issues and finding relevant material to explore further in academic and scientific applications.

Visually, the resulting representation used here is a graph, whose nodes represent points of the data set (individual textual documents) and whose edges represent some type of relationship between the connected nodes. The graph displays and interactions presented in this Section were generated using PEx, the Projection Explorer, a freely available tool for multi-dimensional visualization via projections and point placement strategies with text processing capabilities (see <http://infoserver.lcad.icmc.usp.br/infovis2/PEx>).

An alternative view of the same text maps is the landscape or surface view, where vertices are points and meshes codify additional attributes by mapping scalar fields to geometric or visual attributes (height, color, glyphs, etc.). Surface views in the following text are generated using the Super Spider tool, a recent evolution of its precursor, the Spider Cursor [MLN⁰⁵], that can be used to explore landscape plots.

The visual representation employed here, whose construc-

tion will be described along the text and slides of this Tutorial, can also be the result of various other tools for multidimensional data projection (see Section 4.2, except, maybe, for particularly visual attribute mappings and for the flying quadric in Figure 5).

2.1. Visual mapping of news flashes

The first data set we shall explore was built by gathering RSS feeds of news flashes from the WWW sites of four news agencies: Associated Press, BBC, CNN and Reuters. They represent all the news made available as RSS during approximately 24 hours in April 2006. After some elimination of duplicates, the gathering process resulted in 2,684 news files. For the visual mapping, we have used title and content only, with the goal that similar news (supposedly news treating the same subjects) are mapped to the same neighborhood. For the exploration after mapping, the full file can be used, which contains also the date and the source of the news (the name of the news agency).

The size of the text in an RSS feed (which, for many applications, synthesizes a larger file) is usually quite small, typically 25 to 80 words only. This test case bears resemblance to many applications available today that wish to analyze RSS feeds of all sorts of information (for instance, discussion groups, support groups, article repositories, patent repositories, dictionaries, libraries, and thousands of other such repositories).

Figure 2 shows pictures of two different maps from this data set. In Figure 2(a), which employs a multi-dimensional projection technique called Least Square Projection, or LSP [PNML07, PNML06], the region labeled A is a group of news articles on the Vioxx dispute. Region B centralizes all news on an immigration bill undergoing discussion at the US senate. Group C concentrates the news on a trial related to the Enron case (a particularly important deposition was

about to take place). By coloring nodes according to the source of the news, it is also possible to identify the degree of importance a particular news agency gave to a subject by estimating the number of points with its color. Conversely, the same observation supports identification of the news agencies that dedicated a lot of — or just a little — effort to report a particular event.

Figure 2(b) shows the same map as Figure 2(a), slightly zoomed in, now presenting a set of labels that support identification of the subject mostly tackled by a particular user-selected group of documents. The set of labels in that picture, calculated by a term covariance approach (see Section 5.3.1), identify some of the central events of those two days.

Interpretation of groups in this type of projection is very similar to interpretation of clusters, that is, highly similar data is identified by proximity in the 2D display. In the very middle of the display remain the groups that have been 'unresolved' by the current context, which means that according to the strategy employed for similarity in this domain, it was not possible to distinguish them from the others. Coloring in Figure 2(b) was carried out using a hierarchical clustering method in 2D (in this case k-means with average linkage weighting). It helps distinguish regions of higher density of points (in the 'blue' end of the spectrum) from regions of lower densities (moving towards red as regions become more sparse).

Figure 2(c) shows the visual result of another map of the same data set, built using a new technique for point placement of this type of similarity-based data, called Neighbor-joining Tree (or N-J tree) [CPMT07]. An N-J tree is a type of similarity tree visualization that employs the same principles of phylogeny tree reconstruction [SN87] to reflect the organization of data points (in this case documents) according to their similarity. The interpretation of that type of visualization is slightly different than straight projections. In an N-J tree, very similar (according to the similarity measure employed) data points are gathered in the same and nearby branches of the tree. Proximity on the 2D plane itself does not imply similarity. Interpretation of similarity trees needs, therefore, the branching information to be effective. Branches guide selection and focus also; the neighborhood of a point is extracted searching up and down the tree. Groups of documents with content that is consistent (that is, that are best resolved by the similarity measure employed) are placed at the outer branches. Figure 2(c) uses covariance labels to illustrate the placement of some of the main news in the data set.

Closer examination of the flash news visualizations reveals that many of them are just headlines followed by a short text of the sort 'read full story for details'. Figure 3(a) highlights, in blue, the points where that occurs. Removing those can help unclutter the view, which was done by building a new map after removing those blue points (see Fig-

ure 3(b)). Similarly, other searches on the map can help highlight, select or eliminate specific files and groups of files.

The news data set, after the reduction illustrated in Figure 3, has 2450 news items remaining. The similarity tree by neighbor-joining of that reduced data set is presented in Figure 4(a). That picture also shows a selection of a branch for further examination. The region is zoomed in on in Figure 4(b), and further levels of topic extraction can be observed. We can see in the central part of that picture that the similarity calculations gathered news on various linked Palestinian issues in neighboring branches. These issues varied from formation of the Hamas government to Israeli strikes in Gaza to halting of financial aid from Europe and the United States. The lighter label in Figure 4(b) is the title of one of the news files in that map.

A surface view of a projection-based document map can be built by creating a mesh from the projected points and then using scalar fields attributed to vertices to reflect properties. The result of such a process is illustrated in Figure 5. Various attributes can be mapped to scalar fields, from search results (for instance the number of appearances of a particular expression) to the results of clustering, classification, and categorization of data points. In Figures 5(a) and 5(b) color reflects a 2D clustering (also by k-means) of the points in the projection, employed to help locate pockets of closely projected documents. Figure 5(c) illustrates the capabilities of simultaneous presentation of multiple scalar fields using visual and geometrical attributes. In that picture, height represents the count of the number of times the word 'Bush' appears in the document (higher points are higher counts, naturally). Each of the news sources (four in all) is reflected by a different color. That same information is redundantly reflected on the color of a 'flying' super-quadric that changes color as the user browses through the map. The visual and geometric mappings of the super-quadric can reflect properties of the currently explored point (or focus point). In the case of Figure 5(c), only its color changes, mapping the news agency that published the document in the focus point. Label by title clarifies what the news is about.

Finally, another type of projection, called ProjClus [PM06] is used to map the same news data set explored thus far (see Figure 6). Figure 6(a) shows the graph view with 2D k-means clustering for grouping location and Figure 6(b) shows the mesh view of the same map. In Figure 6(b) color is hierarchical clustering, that is, darker blue regions are regions of largest concentrations of points, down to red for sparser regions. ProjClus tends to spread the clusters around more, which is confirmed by larger regions with the same top density. Although that property sometimes confuses the boundaries of clusters, ProjClus still bears good 'proximity by similarity' precision and can help estimate the local density of a region since it diminishes cluttering compared to LSP.

Both projections thus far employed in the illustrations are

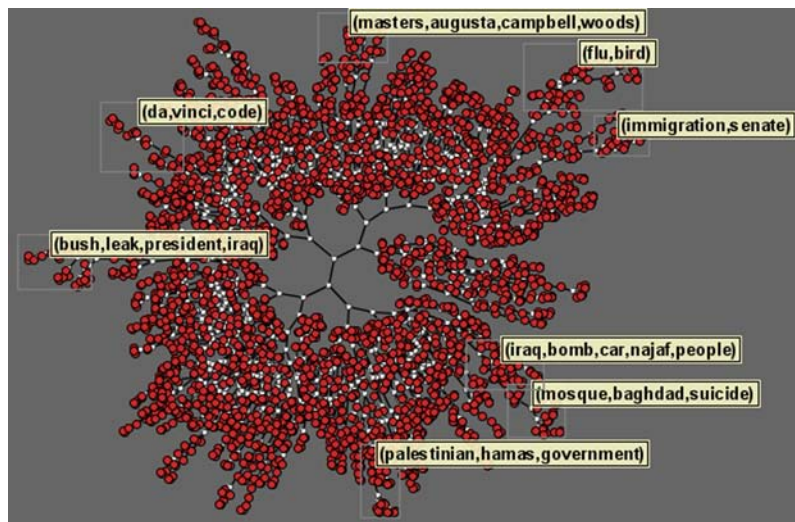
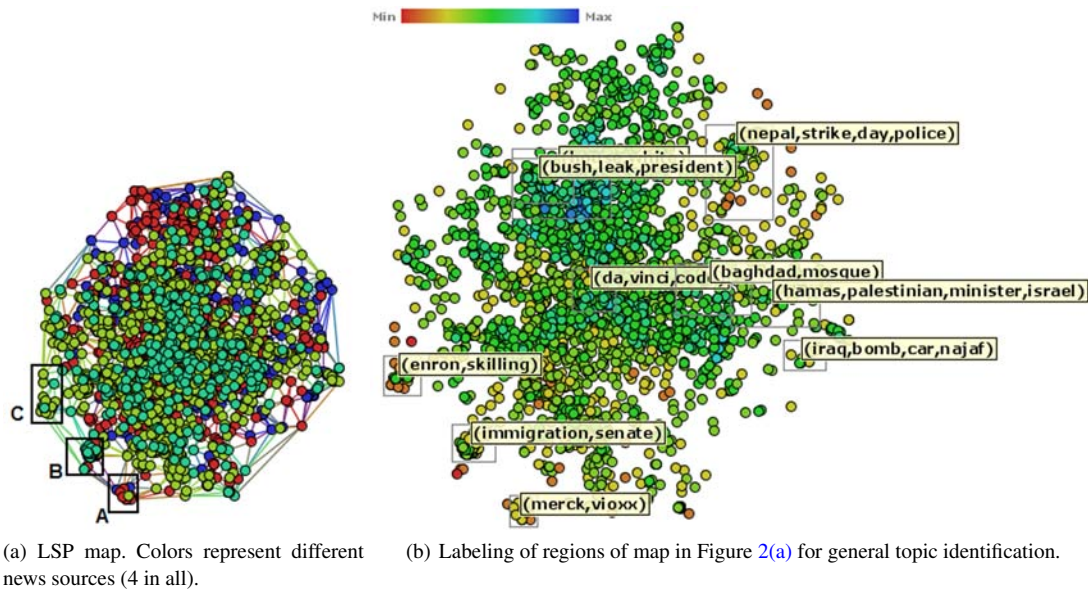


Figure 2: Document maps of 2,684 news flashes collected for approximately 24 hours in April 2006.

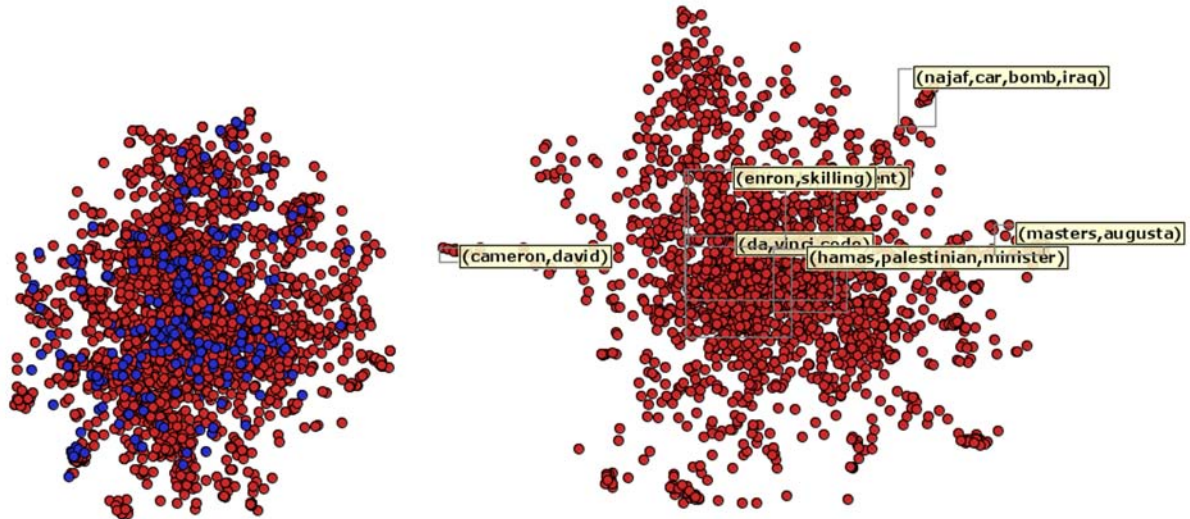
reasonably fast, leading to an environment where simultaneous examination and coordination of various maps is feasible. Similarity trees have higher computational cost but it is still manageable for data sets whose size is in the hundreds of points, even in the thousands, if similarity is pre-calculated and stored. Neighbor-joining algorithms are typically $O(n^3)$ or $O(n^2)$ at best.

2.2. Visual mapping of scientific papers

The second data set to be illustrated here is composed of 676 papers in four areas of knowledge: Case-based Rea-

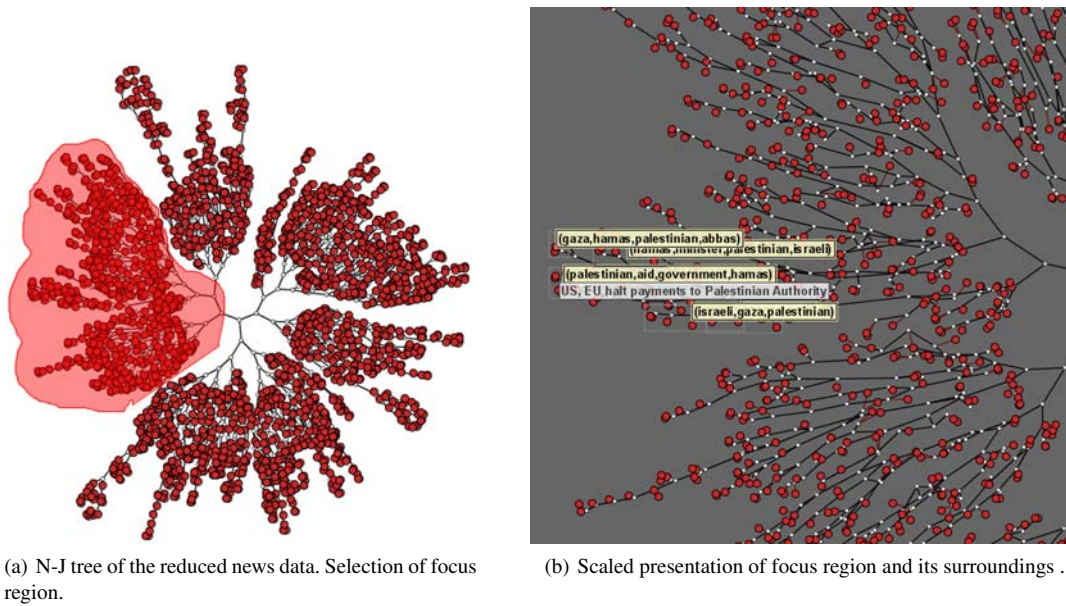
soning (CBR), Inductive Logic Programming (ILP), Information Retrieval (IR) and Sonification (SON). CBR and ILP documents were extracted from Lecture Notes in Computer Science (LNCS) in those subjects. IR and SON papers were retrieved from internet repositories. They are pseudo-classified according to their source. The maps were built using papers' titles, authors, affiliations and references.

In addition to those documents, six others were added to the data set. These were papers of work published by members present and past of our research group; the goal was to evaluate their local and global positioning on the map. From those, five papers were highly related. They refer to



(a) Highlighting news with words ‘full’ and ‘story’ on map of Figure 2(a) (b) New map without news highlighted in Figure 3(a). 2450 news files remained

Figure 3: Using search and corpus manipulation to eliminate unwanted files



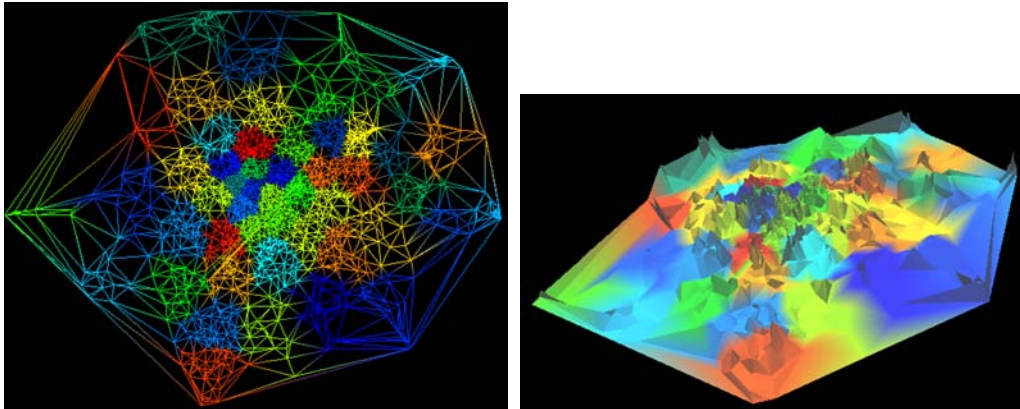
(a) N-J tree of the reduced news data. Selection of focus region.

(b) Scaled presentation of focus region and its surroundings .

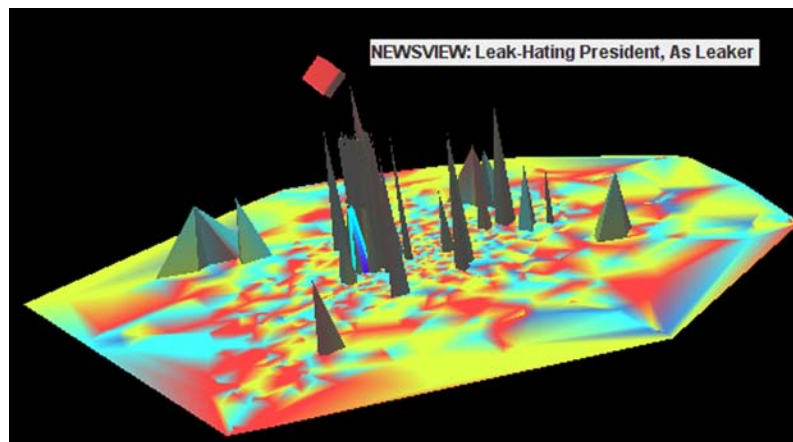
Figure 4: Focusing on a part of a Map for further exploration

the evolution of a sonification system and its corresponding user evaluation. Two of them refer to the first version of the system and the other three to its newest version. The last paper was also published having as authors members of our research team, but on a completely different subject (a technique used for image segmentation). Let us call those six additional papers the ‘intruders’.

Figure 7 shows general and specific views of the papers data set, mapped by LSP. It can be seen in Figure 7(a), both by coloring and by labeling (each color here represents a pseudo-class), that the projection technique was capable of separating well this heterogeneous document collection by their subject areas. Moreover, subjects with an established and more focused body of techniques, such as CBR and ILP



(a) Mesh generated from a Delaunay triangulation of news (b) Surface view of news set. Color and height codify the map in Figure 3(b). Color is k-means clustering of the projected points.



(c) Surface View. Color reflects the source of the news; height codifies the number of appearances of the word 'Bush' in that particular text; the color of 'flying' geometry on the display confirms the source of the news (i.e., the news agency here represented in red) of the focus point; the focus is highlighted by its incident edges colored in blue; label is title of the focus point.

Figure 5: Surface representation and exploration of content by visual attributes.

tend to be displayed less sparsely than a newer more diverse area such as Information Retrieval. In the various regions where points of different colors appear close to one another, further examination reveals that there was a high content correlation that lead to that proximity. Most of the time they are applications of techniques in those areas (CBR, ILP, IR, SON) for particular goals (such as modeling, manipulation, or study of particular data sets such as biological sequences).

To check that capability further, it can be seen in Figure 7(b) that all except one of the intruder papers were mapped in the same group. The five sonification papers share a very close neighborhood within the general sonification group, as would be desired. The sixth, unrelated paper, joined another separate group, that seemed to gather together

papers from different subject areas. Closer examination by focussing on that particular cluster and loading the documents represented there (see Figure 7(c)) shows that these papers deal with various techniques (in all subject areas) designed to treat imaging tasks (such as retrieval, manipulation and segmentation). In fact all papers in the paper data set dealing with imaging techniques were placed in that group.

Connections amongst nodes on the graph can carry various meanings. For instance, they may reflect neighborhoods within the data set. Figure 8 illustrates the display of similarity relationships also as edges on the graph. In Figure 8(a) documents are connected according to the similarity measure employed for the creation of the map. Each point is connected to its next neighbor in the similarity matrix. It can be

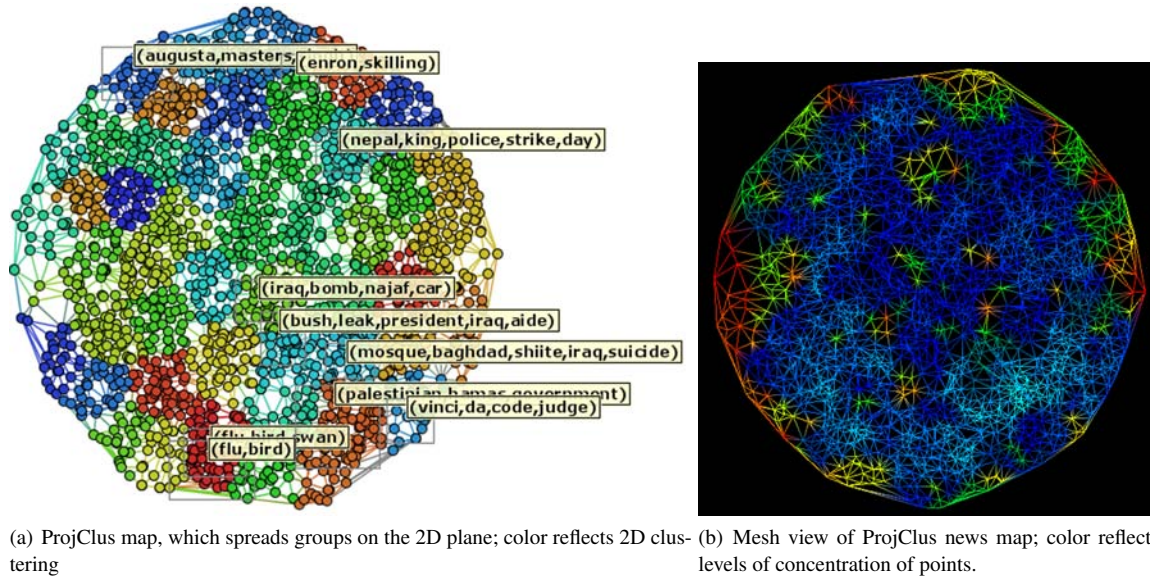


Figure 6: Map by ProjClus of the reduced news set.

seen that there are a few points whose next neighbors are not within the same general group. That is to be expected, since the main groups are not at all disjunct as far as content similarity is concerned. Still, the small number of 'crossings' outside the clusters indicates the good degree of grouping presented by the projection. New connections can be added. For instance, each point can be connected to its two nearest neighbors in the original data space (see Figure 8(b)); in this case more connections across groups appear, and they can reveal further associations apart from those already revealed by grouping and first neighbors, adding to the possibilities of exploration.

Many other relationships could similarly be mapped to edges on a map. Focussing on parts of the map shows in further detail the subjects within groups. Detail of two groups of the map (see Figure 8(d)) reveals that, within the general sonification group, there are separate groups for audio processing (including retrieval), computer music, sound in user interfaces, data sonification, and sonification for the visually impaired.

Display on the 2D plane of multidimensional data can be done using a number of techniques. As well as projections, other point placement strategies can be employed. Sometimes it is useful to arrange (or rearrange) a projection according to the connections amongst points. That is illustrated in Figure 9(a) and 9(b). Other important issues in document map exploration are topic extraction and visualization, which help the user find interesting groups of documents to explore further. Figure 9(c) illustrates that using a proper technique (selective generation of association rules, in this

case), further detail of topics approached by documents can be viewed.

Finally, we map the scientific paper data set using N-J trees. Since the tree reflects an organization of the similarity measure, it reveals subtleties not easily identifiable in projection layouts. For instance, in Figure 10(a) and 10(b), just by having two different approaches to display the same tree, it is possible to locate different levels of grouping and separation of groups. Additionally, in the same pictures it can be seen that the positioning of the 'intruder' papers in the same neighborhood happens in two levels for those dealing with sonification. The two papers referring to the first version of the sonification system are gathered in the same branch while the others are together in another branch nearby. Figure 10(c) shows an N-J tree map built from another similarity measure, in this case Normalized Compressed Distance (NCD) [TMP07, CV05], see below for more details. Overall NCD gives similar separation of greater groups for this same data set. However, close examination shows that not all relationships are maintained. For instance, in that map all the intruders were, according to that NCD, deemed more similar than using vector representation. This type of presentation gives a good reflection of how a particular similarity measure behaves with a data set.

The cases shown here illustrate the use of projection-based displays together with mining techniques (such as clustering, covariance term extraction and association rules extraction) in an integrated manner, with combined available technology to build maps for exploration of text collections. These and other tools put together can support improvement

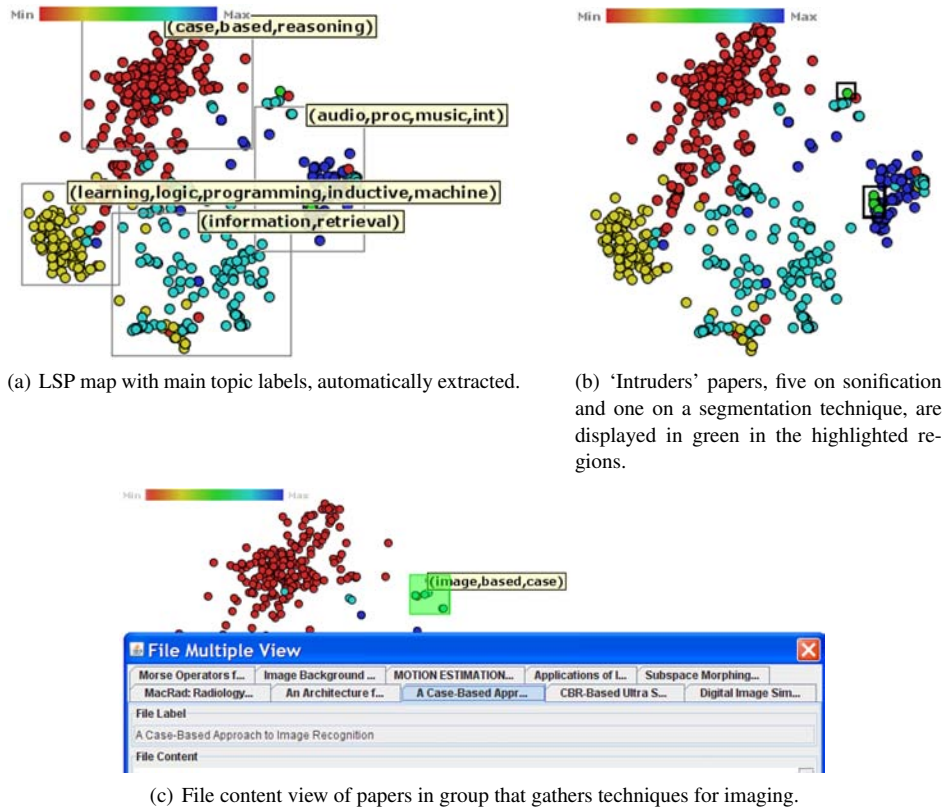


Figure 7: Map of scientific papers in four classes, CBR, ILP, IR and SON, using LSP.

in examination of larger text collections without extensive reading.

The next section starts detailing concepts and techniques related to the construction and exploration of such maps.

3. Basic Concepts

3.1. Text processing and information retrieval

3.1.1. Introduction and topics

We summarize the fundamentals of text processing and information retrieval here. The main topics to be explored are:

- Modeling for information retrieval
- Retrieval evaluation methods
- Query languages and operations
- Text and multimedia languages and properties
- Text operations
- Indexing and searching

Whereas this include non-text aspects, such as multimedia languages and properties, we leave those here to provide a more comprehensive picture of the subject.

3.1.2. Motivation

Information Retrieval (IR) has become a major factor in today's information-centered world. Whereas in the past, only very technical people used to interact with data sources — actually, mostly databases — today lay people used data sources on a daily basis. Every Web search, every Web-based purchase, and many other application on- and off-line, all focus on data sources and data bases, whether the user is or is not aware of it.

3.1.3. Information versus data retrieval

3.1.3.1. Data retrieval : Data retrieval interacts with databases, which are sources of structured data. Databases do not usually contain broad information about a specific "subject or topic", but, rather, store specific data in structured tables, such as catalogs, populations data, and the like. Retrieval from a database requires some knowledge of a query language — the most prevalent variations on SQL, the Standard Query Language. Data retrieval will only succeed if the database includes specific instances of precise query terms. A slight misspelling can make the difference between hit and miss, between success and failure.

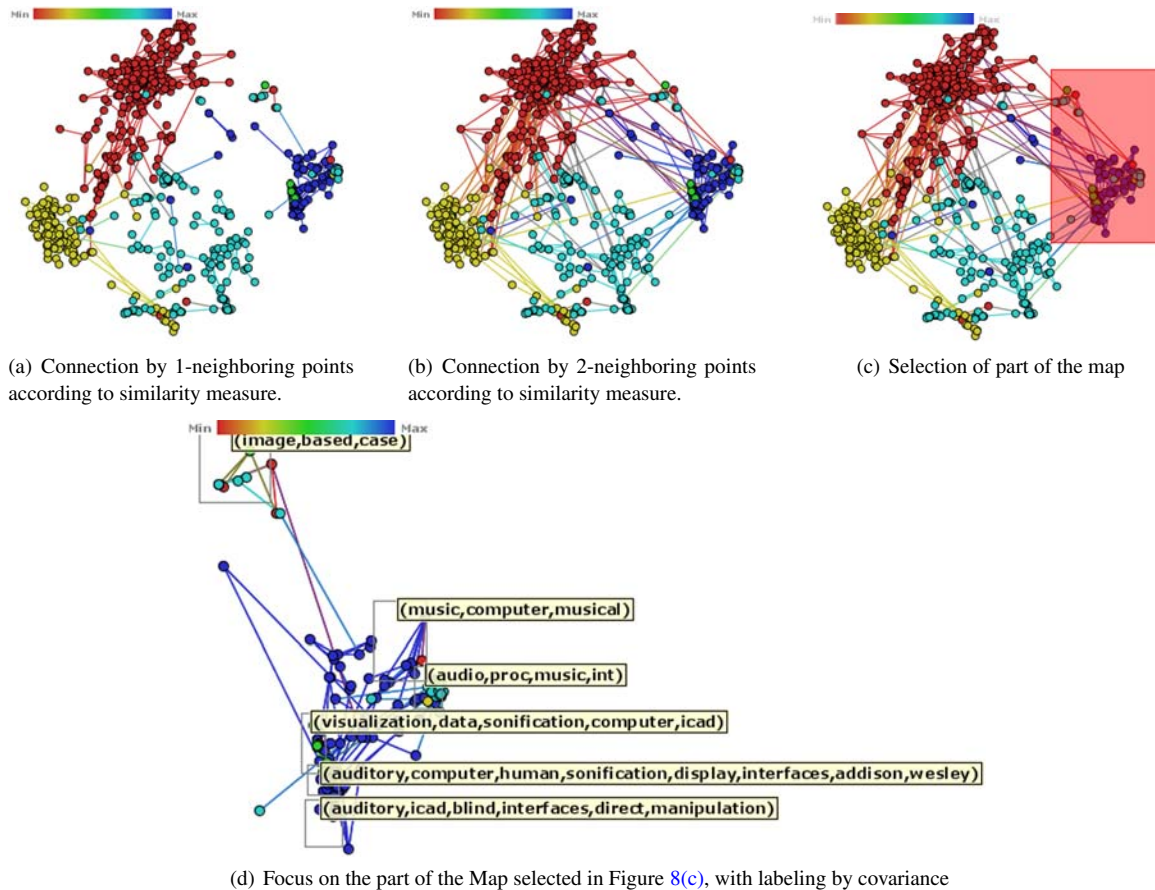


Figure 8: Maps of scientific papers with connection representing similarity in the original space.

3.1.3.2. Information Retrieval (IR) : IR, on the other hand, focuses on retrieval from unstructured documents that are related by topic or subject. Query terms are not necessarily expected to be present in their exact form in a document for it to be returned in the result list. Instead, IR systems attempt to interpret the contents of information items — mostly, text documents in a collection. This interpretation involves a comparison between the entered query and the documents in the collection, computation of a “difference” or “distance” measure, and ranking of the returned documents according to their relevance to the user’s query.

3.1.4. Information retrieval at center of stage

Information Retrieval has been utilized by librarians and other information professionals for close to three decades. However, with the penetration of the World-Wide Web, it is now one of the most important tools a wide variety of users utilize in their day-to-day interactions with information. The Web has become the single largest repository of, as well as the unified interface to most of the information people have to interact with. However, the Web presents a

significant obstacle to effective and efficient information harvesting, namely, the absence of a well-defined underlying data model for Web-based information. This typically leads to information definition and structure that is frequently of low quality.

3.1.5. Basic concepts of IR

The IR process starts with a user’s need. The user’s initial task is to translate her information need into a query, specified in a language that is provided and understood by the system. We have all grown accustomed to entering multiple daily queries into our favorite search engine. If the user’s query is entered into an IR system, she typically would specify a set of “key words”, which are expected to convey the semantics of the information needed. Using a more rigid data retrieval system, a much more rigorous query expression is required, such as a regular expression, which contains constraints to be satisfied by objects in the answer set. In both cases, the user searches for some useful information executing a retrieval task.

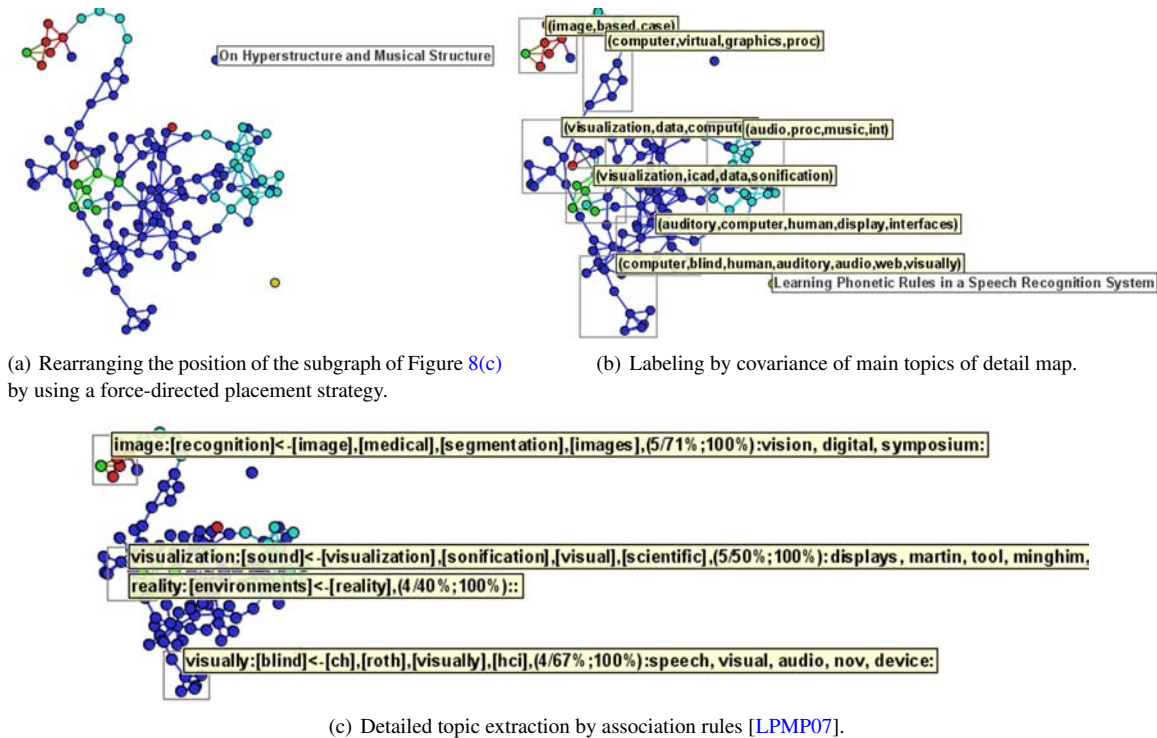


Figure 9: Exploration of a sub-graph with rearrangement and topic extraction and visualization.

A schematic overview of the basic iterative retrieval process, the logical view of a document, and a more detailed picture of the retrieval process, can all be found in the accompanying slides.

3.1.6. A brief history of Information Retrieval

Here, we summarize the history of Information Retrieval, covering the following topics:

1. Early developments;
2. Information Retrieval in the library;
3. the Web and digital libraries.

3.1.7. Early developments

For roughly 4000 years, people have been organizing information to be able to later retrieve and use it as they need. One of the most common examples is the familiar Table of Contents found in many books and other text documents. After Gutenberg's invention of the moveable printing press in 1455, which has made books available and affordable to the general population, rather than just to the wealthiest few who could afford them, the volume of information people have acquired grew beyond just a few books. This, in turn, triggered the development of specialized data structures for faster access to stored information. For example, an old popular data structure for faster IR still found in many books is the index.

An index is a collection of selected words and concepts that serve as associated pointers to related documents and other information sources. Indices are at the core of any modern IR system. They provide faster access to data and they speed the task of query processing. For centuries, indices (aka indexes) were created manually, according to categorization hierarchies. Today, libraries still use categorical hierarchy — which have usually been conceived by human subjects from the library sciences field — to classify volumes and documents.

Computers have led to the automatic construction of large indexes, which has led to another view of the retrieval problem, one that is much more related to the system itself than to the user's needs.

We, thus, face two views of the IR problem, a Computer-centered view, and a human-centered one.

3.1.8. A computer-centered view of the IR problem

As far as computer systems are concerned, the IR problem has the following components:

- build efficient indexes;
- process user queries with high performance; and
- develop ranking efficient algorithms.

All, with one goal in mind: to improve the 'quality' of the retrieved answer set .

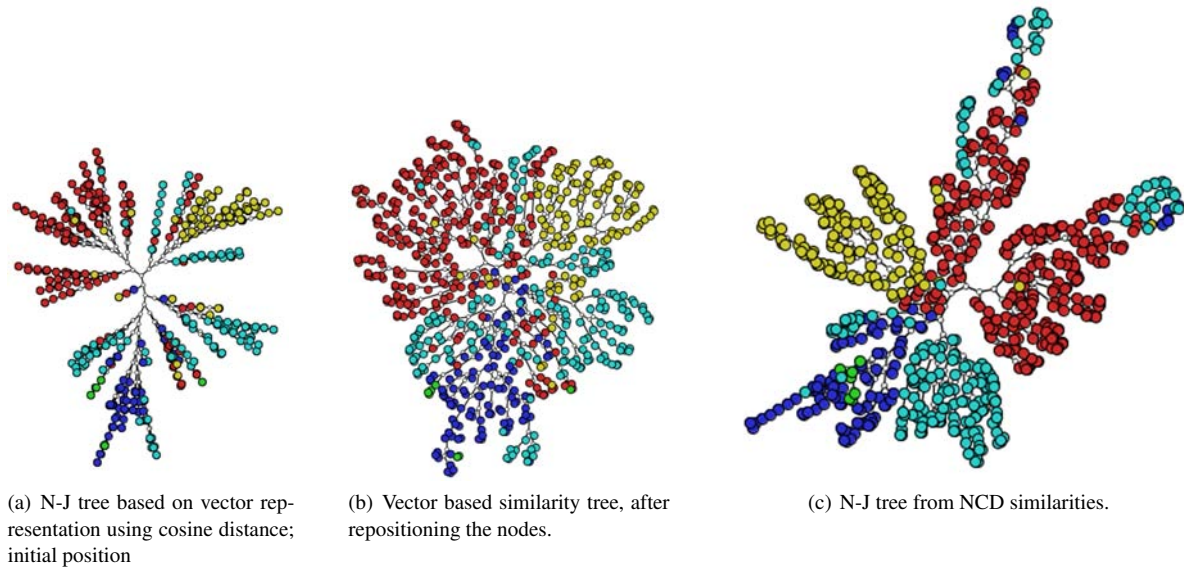


Figure 10: Similarity trees for the scientific papers data set.

3.1.9. A human-centered view of the IR problem

From the human user's point of view, the IR problem can be summarized as:

- Study the typical behavior of users;
- understand their main needs; and
- determine how such understanding affects the organization and operation of retrieval systems.

From this point of view, a keyword-based query processing might be seen as unlikely to yield a good solution in the long run, leading to the quest for other, more promising alternatives.

3.1.10. Information Retrieval in the library

Libraries were among the first to adopt IR systems. Those systems were developed initially by academic institutions but later by commercial vendors started offering more refined tools and systems. We observe first, second and third generation features in those systems.

3.1.10.1. First generation . The first generation of library IR systems was basically an automation of previous technologies, mostly card catalogs. They allowed searches based on author name and title only.

3.1.10.2. Second generation . Here, increased search functionality added search by subject headings, keywords, and some more complex query facilities.

3.1.10.3. Third generation . More recently deployed, these systems focus on improved graphical interfaces, elec-

tronic forms, hypertext features, and open system architectures.

3.1.11. The Web and digital libraries

The Web has become the primary source of all kinds of information. With the spiral growth of the amounts of information available on the Web, search engines have become among the most important Web tools — and assets. Today's search engines continue to use indexes that are very similar to those used by librarians a century ago. So what has changed?

We observe three dramatic and fundamental changes: First, access to various information sources has become a lot cheaper, reaching wider audiences than ever possible before. Second, advances in digital communications have led to easier, cheaper, and thus greater access to networks, allowing distant, quick access to vast amounts of information throughout the world. And, third, the freedom to post whatever one would like to, has caused the Web to become ever more popular. For the first time in history, most people have an almost free access to a large publishing medium, making the Web (and modern digital libraries) a highly interactive medium, facilitating the exchange of messages, photos, documents, software, video; making interactive chatting convenient and at low (or no) cost, cause overall a fundamental shift in the current communication and information flow paradigm.

3.1.12. The future: three main questions

We are still not through; we do not have all the knowledge, all the answers. The following still need to be addressed:

First, despite the high interactivity, people still find it difficult — impossible at times — to retrieve the information that would be relevant to their information needs, leading to the question “which techniques have the potential to yield retrieval results of higher quality?”

Second, ever increasing demand for access is causing the need for quicker response to be more and more a pressing factor. We therefore need to find out which techniques can support faster indexes and smaller query response times? And,

Third, the quality of the retrieval task is greatly affected by the user’s interaction with system, so “how can better understanding of the user’s behavior affect the design and deployment of new information retrieval strategies?”

3.1.13. Practical issues

The following issues have also great influence on the entire IR problem and solution set.

- Electronic commerce, a major trend on Web;
- security;
- privacy;
- intellectual property rights and publisher responsibilities;
- internationalization; and how to deal with multimedia (images, sound, video, etc.)

3.1.14. Overview of the rest of IR topics

There are many components to IR. We list here the major topics; these remain, however, outside the scope (and time and space limitations) of this tutorial. The reader is encouraged to explore these further. A very comprehensive treatment — and, thus, a good place to start — would be [BYRN99].

1. IR modeling;
2. retrieval evaluation;
3. query languages and operations;
4. text and multimedia languages and properties;
5. text operations;
6. indexing and searching;
7. parallel and distributed IR;
8. user interfaces and visualization; and
9. searching the Web

3.1.15. Text preprocessing for IR, mining, and visualization

In order to reduce the space dimension and to allow refining the text model, text is processed prior to extraction of the vector representation. This process typically involves three steps: (i) removing stopwords, i.e., non-informative words, such as articles, prepositions and such, plus any words known to lack relevance to the context; (ii) stemming, which reduces words to their radicals (e.g., ‘motivation’, ‘motivate’, ‘motivating’, would be all reduced to ‘motiv’); and (iii) frequency counting, so as to remove terms that occur too sparsely or too often and hence have little differential

capability. Setting suitable frequency thresholds for discarding terms typically demands user knowledge and interaction. Corpus vector representation enables handling the map generation problem as one of mapping objects defined in a high dimensional space into a 2D (or 3D) visual representation space. Clustering and projections are typical approaches to handle the problem. Determining vector similarity — a necessary step for both processes — requires defining a metric in the high dimensional vector space that allows computing distances between vectors. The distance between any pair of document vectors is a measure of their proximity, or semantic similarity. Distances for all pairs can be stored in a triangular distance matrix of dimensions $n \times n$, where n is the number of documents.

An alternative to the vector space model has been proposed based on Kolmogorov complexity approximation [TMP07], called Normalized Compressed Distance (NCD). NCD computes a distance between a pair of documents straight from their textual content, rather than from the intermediate vector representation. The Kolmogorov complexity of a string x , denoted $K(x)$, is the size of a description of x produced by an optimal specification method. The conditional Kolmogorov complexity $K(x|y)$ of x with respect to another string y can also be defined as the amount of information that x does not have about y .

An information distance between two given strings can be specified based on their conditional Kolmogorov complexity. Though the problem of evaluating a string’s Kolmogorov complexity is non-computable, a solution can be approximated through compression; this yields the basis for the definition of NCD. We include a technical report in Appendix A on a formulation of NCD for text collections at the end of the Tutorial notes.

Other alternatives for pairwise similarity calculation, particularly those based on information theory, have also being studied recently [AF03].

3.2. Data and text mining

3.2.1. The knowledge discovery process

Data Exploration is the process of searching and analyzing databases to discover implicit but potentially useful information, mostly for the purpose of supporting the decision making process. The goals of data exploration are:

- Convey information;
- discover new knowledge; and
- identify structure, patterns, anomalies, trends, and relationships.

Major Data Mining Tasks, such as, summarization, association, classification, prediction, clustering, and time-series analysis use major techniques, including Linear Regression Trees, Non-Linear Regression, MARS, Naive Bayes, K-Means and K-Median, Neural Networks, Association Rules,

Decision Trees, Principal Component Analysis, Support Vector Machines, and Genetic Algorithms. These, in turn, are based on statistical tools, such as, Missing Value Imputation, Normalization techniques, Error & Variational Analysis, and Confidence Estimates, to list just a few.

3.3. Projection techniques

Data sources have increased substantially both in size and complexity, but extracting useful information from them is still a challenge. One measure of data complexity is the number of attributes associated with each instance of data. Consider, for example, data from a demographic census: a data instance records attributes such as age, sex, education, occupation, income, and so forth. Considering each data attribute as a data dimension, if we have m such attributes each data instance can be interpreted as an m -dimensional vector placed in an m -dimensional definition space.

In traditional statistical analysis, data instances with three or more dimensions are known as multivariate or hyper-variate data. In Information Visualization such data is usually referred to as multidimensional. Conventional methods, such as scatter plots or bar charts, normally employed to assist data interpretation, are not directly applicable to multidimensional data. Moreover, identification of patterns and models grows more difficult as dimensionality increases, and lack of proper representations can severely impair interpretation.

A common way to handle dimensionality is to reduce the number of dimensions, so that strategies that are known to work well with low-dimensional data can be applied. *Multidimensional Projection* techniques are one example of such a strategy. A multidimensional projection technique typically maps the data into a p -dimensional space with $p = \{1, 2, 3\}$, while retaining, in the target space, some information about distance relationships among the data items in their original definition space. In this way, a graphical representation can be created to take advantage of the human visual ability to recognize structures or patterns based on similarity, such as clusters of elements.

Formally, let $X = \{x_1, x_2, \dots, x_n\}$ be a set of m -dimensional data, with $\delta(x_i, x_j)$ a dissimilarity (distance) measure between two m -dimensional data instances, and let $Y = \{y_1, y_2, \dots, y_n\}$ be a set of points in a p -dimensional target space, with $p = \{1, 2, 3\}$ and $d(y_i, y_j)$ a (Euclidean) distance between two points of the target space. A multidimensional projection technique can be described as a bijective function $f : X \rightarrow Y$ that seeks to make $|\delta(x_i, x_j) - d(f(x_i), f(x_j))|$ as close to zero as possible, $\forall x_i, x_j \in X$.

3.4. Visual representations: graphs, surfaces, volumes, triangulations

Many visual representations have been utilized to represent data. Among the most commonly used ones one finds

maps, graphs and networks, surface- and volume representations. The use of maps is so widespread that it is not necessary to discuss them, as every one knows their utilization for geographical applications. The concept has been used also to represent data in which some form of “proximity”, whether in physical space or as determined by other measures, is needed to be presented.

Whereas maps are particularly useful to represent physical proximity or its conceptual equivalent, graphs and networks have been primarily utilized where the relationships among participants are to be emphasized. In these types of situations, measurable physical is not important; what is important is how nodes are related to each other. For example, the communication patterns among members of an organization can be nicely visualized by a graph, in which a link between two nodes represents the communication between them. A directed link can show who initiated the communication. A network of such communications can quickly show influential figures, such as “authorities” (those who are considered knowledgeable about a variety of subjects) and “hubs” (those who are known to seek when a question needs to be answered). The mapping of authorities and hubs have been the cornerstones of the original “page rank”, the algorithm that provided the initial implementation of Google’s search engine. In addition, graphs and networks are common representations for processes in which transitions occur between states, under certain conditions.

Spatial data — and, at times, data that can be mapped to a spatial representation — have been often represented by volumes, or their surfaces. Volume representations preserve the values of data throughout the volume, whereas surface representations discard internal values and only preserve values of the external surfaces selected to be displayed.

The two most crucial aspects of either are the data structures utilized to represent the data, and rendering techniques to display the data on a screen. Multiple data structures have been proposed for storing data of three (or higher) dimensions. Among them, in addition to the straightforward cartesian coordinates, are various hierarchical representations, including trees of various type. The discussion of such data structures are beyond the scope of this tutorial; the reader is referred to the standard data structure literature for details.

Surface representations have included various triangulation techniques, as well as cuberille approaches. Triangulation techniques approximate the surface with a mesh of triangles that attempt to follow the surface as closely as possible. Cuberille techniques try to approximate the surface with a mesh of squares rather than rectangles. Some use graph concepts to implement optimal coverage.

The display of three- or higher-dimensional data on a two-dimensional display requires projection first. Surface and volume rendering techniques have been covered extensively in the computer graphics and visualization literature. The

reader is referred to this literature to familiarize her/himself with the technical details entailed.

The accompanying slides provide some examples of various of these techniques.

4. From Visualization to Visual Text Mining

4.1. Visualization techniques for multidimensional data

“One picture is worth a thousand words” is an old cliché, but it is based in the fundamentals of human perception and processing of information. Sixty percent of all input for decision making comes into the brain of a normal-visioned person from the visual system. It has been demonstrated that, in the event of contradicting information between the visual input and that of another sense (e.g., tactile), the visual stimulus with “win” over the other one, even if the other sense is providing correct information and the visual input is erroneous!

Graphics and Visualization are meant to help the user see (understand), remember, compute, analyze, discover, enjoy, and much much more.

Multiple goals are served by different types of visualization tasks, and the appropriate techniques. The most common visualizations used are for the purpose of presentation of known information. Here, facts to be presented are known (though they may not represent the truth), the visualization process is to choose and tune the appropriate visualization technique, and the result is a high-quality visualization of the data and analysis to present facts (often without the author’s presence). Confirmatory analysis visualization starts with some hypotheses about the data, proceeds through a goal-oriented examination of the hypotheses, and results in a visualization of the data to confirm, accept, or reject the hypotheses. Finally, exploratory analysis starts with no hypotheses about the data, proceeds with an interactive, usually undirected search for structures, trends, patterns or anomalies, and yields a visualization of the data to lead to some hypotheses about the data.

All of these utilize a wide variety of technique, some are pure visualizations, some are integrated with analysis, all utilizing to one degree or another an assortment of interaction tools, which help the user control the process and interact with it to yield optimal results.

Some pure visualization techniques include 2D and 3D Scatterplots, Matrix of Scatterplots, Statistical Charts, Line and Multi-line Graphs, Parallel Coordinates, Circle Segments, Polar Charts, Survey Plots, Heatmaps, Height Maps, Iconographic Displays, RadViz, PolyViz, and many more.

Among those that are integrated with analysis one finds Projection Pursuit, Dimensional Stacking, Sammon Plots, Multi-Dimensional Scaling, Principal Component Analysis (PCA) and Principal Curves, and Self Organizing Maps.

Interactions include Selection, Probing, Querying, Grand Tours, and Non-linear Zooms.

It would require hundreds (if not thousands) of pages and hours to discuss all of these in even a moderate amount of details, so we will limit our discussions here to only the very basic concepts, and to a few methods that are particularly relevant to our main focus, Visual Text Mining.

4.1.1. The visualization problem

The visualization problem can be summarized as: Massive amounts of data from various sources, including databases, simulations, sensors, decision systems, and more; limited screen space; and little knowledge about the human perceptual system and the process of information transfer.

Visualization is a method of computing. It transforms the symbolic into the geometric, enabling researchers to observe their simulations and computations. Visualization offers a method for seeing the unseen. [MDE87]

Visualization now includes other data representations, such as auditory, haptic and tactile, potentially more in the future.

4.1.2. Classifications of visualization techniques

We are primarily interested in those visualization techniques that are suitable for multi-dimensional data sets, as visual text mining deals in high dimensional data.

Many classifications have been proposed for multi-dimensional data visualization techniques.

4.1.2.1. Point-based Point-based techniques map the dimensions of a data record to the attributes of a point on the display, including its size, shape, color, motion, and — occasionally — sound at the point. The most recognized point-based technique is the scatterplot. Scatterplot-based visualization techniques include OmniViz Galaxy (OmniViz, Inc.), Temple MVV Graphics [MGTS90, MTS91], [CC92], and Splat Visualizer and Scatter Visualizer (from Purple Insight).

Because the traditional scatterplot is essentially a two-dimensional data technique, various approaches have been proposed to extend scatterplots to higher dimensions. These approaches include layout extensions, either via matrices or radially, and shape extension, utilizing icons and glyphs.

Scatterplot Matrices provide an xy layout of k -dimensional data in a total of $k(k-1)/2$ scatterplots, each showing two of the k dimensions against each other. Clearly, as k grows, the number of scatterplots grow too. E.g., for $k = 100$ the number of scatterplot will be 4,950. The concept has even been extended to scatterplot cubes, following the same idea.

RadVis [Hof99] is a point-based technique that organizes

data dimensions around a circle. Data dimension values function like springs, exerting forces over a data record in various directions. The result is a collection of point clouds, clustered based on dimension values.

Hyperslice is a matrix of k^2 slices through a k -dimensional data set. Slices can be determined interactively [vWvL93].

4.1.2.2. Line-based Line-based techniques include bar charts, line graphs, parallel coordinates.

4.1.2.3. Hierarchical, graphs, trees These methods utilize hierarchical structure, networks, graphs, and trees to present data that is inherently suitable for such structures. Among the best known technique one finds Eick's SeeNet [BEW95], which utilizes spoke, helix, and sphere layouts to plot networks of nodes and their relationships; SeeNet 3D [Eic96], which is an extension of SeeNet to three dimensional rendering; SeeNet ArchView, a member of the SeeNet family, which uses arches as the visual primitives; MBone [MB94]; and DBMiner [HCC*97].

4.1.2.4. Iconographic displays Iconographic displays are perceptually driven displays. They generalize the notion of a pixel to that of an icon (or a glyph). By doing so, they increase the number of parameters "displayed". Icons' visual, color, and auditory attributes are mapped to data parameters, thus making the icon "data-driven". The icons are displayed en-masse and thus harness the perceptual powers of the human early visual system. When presented, icons are available for user interaction. These methods have been successfully used in data fusion of multiple parameters / dimensions.

Among the best known iconographics/glyph systems are Chernoff faces [Che73], Andrews' graphs, Stars or circular plots (Ward et al., various), Stick-figure icons [PG88], Color and Texture [H.91], Beddows' Embedded blocks, Smith's Sound (1989-90), and numerous other examples dated during the nineties.

The accompanying slides provide many examples of these various visualizations.

4.2. Visualization techniques and systems for handling document collections

Several different techniques for visualization of textual results from Web and other searches have been proposed ([ABY03, LA00, SCL*99, BY96]). While these techniques are capable of displaying large document bodies, they tend to make the location of specific relevant reading material difficult. We focus here on complementary tools to support mapping of documents in a way that helps locate neighboring similarities between individual text documents and groups of documents. We therefore assume a pre-filtering task that reduces the universe of targeted documents to a few hundreds

(maybe thousands) in a few areas of interest (not necessarily pre-determined).

Many techniques for text document analysis and visualization exist. They usually search for a representation of the content of an individual document or text fragment (e.g., [MWBF98], [RES98], [RES99]), of document collections (e.g., [LA00], [BCG*99], [Wei01]), or of themes approached in text documents (e.g., [HHWN02], [Wis99], [WTP*95]) in order to meet these goals.

The process of text document analysis and visualization usually involves three phases: (i) pre-processing; (ii) dimension reduction; and (iii) attribute mapping to (at least) a visual representation and presentation.

The pre-processing phase takes as input the document collection to be analyzed, and produces an intermediate form, usually based on the vector space model [Sal91] whereby documents are represented as points in a vector space. In this representation each document is represented by a vector whose dimensions are terms (n -grams). The vector coordinates are the weights of the terms based on their frequency of occurrence in the document. Typically, dimensions reach the thousands even for small to medium-sized databases. Transformation of a document collection into a vector space is preceded by elimination of non-influential words (such as stopwords), reduction of words to their radicals (stemming), and some sort of frequency counting (various exist).

The second phase, dimension reduction, typically involves removal of words that are either too frequent or too rare in the collection, and clustering dimensions to generate new 'combined' attributes.

The most common way to extract structure from a text document collection is by applying some dimensional reduction technique over the resulting vector representation. Systems that implement such approaches include those based on Multi-dimensional Scaling (MDS), Principal Component Analysis (PCA) or Latent Semantic Analysis (LSA), which work with statistical measures for subspace reduction, as well as Self-Organizing Maps (SOM), which employ neural computation ([Wei01], [Wis99], [BCG*99], [KHLK98], [WTP*95]).

In the third phase, such techniques can be used to reduce the dimension to two and then plot the data onto 2D space.

However, multidimensional reduction techniques can cause some difficulties, such as [HWR03]: 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 an inherent discretization problem associated with techniques such as SOM, by which individual documents in groups are not distinguishable. For our goal here, dimension reduction can pose an additional problem: when used to display the results in 2D, the mappings to subspaces may define groups of 'similar

documents', but it is not possible to locally relate neighboring documents. Previously [LMMP06], LSA has been successfully applied to generate document maps with high local content relationships, but the high computational cost remains a problem as well as the handling of vector transformations. Minghim et al. [MPL06] have proposed faster mapping approaches that possess the ability to associate documents by similarity, based on dimensional reduction by fast projections (such as Fastmap [FL95] and Nearest Neighbor Projection (NNP) [TMN03]). The gain in processing time is attained by using these projection techniques, which provide an initial point placement that is prone to speed up force-based improvement schemes. Those too are based on the vector representations. Those projections provide an initial placement on the visualization plane, but also require further use of point placement strategies [TMN03] in order to recover information lost during projections.

Point placement strategies, including force-based point placement, have been used before to generate document displays [Cha96, AKS*02a, GSAK04]. They can avoid, partially or completely, the extensive calculations needed in dimensional reduction techniques by starting with a semi-random point placement and re-adjusting their position based on attraction by similarity. However, our experience trying to find relations in their application to text content has been that the precision of the placement is limited as applied to our goals for these maps.

Other major techniques to visualize inter-document similarities are document networks ([FFW91], [TC89]), spring embeddings ([CC92], [SA98]), and document clustering ([AOL93], [HP96], [LC96], [ZE98]). Of those, only document clustering is sufficiently fast and produces intuitive results to warrant consideration ([ZE00], [HP96], [Zam98], [ZK05], [CHR03], [RK04]). It is often employed in combination with dimensional reduction and SOM ([IR01], [Wei01], [LA00], [Koh97], [Lin91], [Mer97]).

Document clustering techniques provide a way to relate documents to each other, as well as to a set of topics. A document is assigned to a cluster if it contains the contents of the cluster's topic, and is generally similar in contents to that topic and to the other documents in the cluster. Documents in the same cluster are expected to be more similar to each other than to documents in other clusters [ZK05]. Since documents may contain multiple topics, some approaches allow a document to be assigned to more than one cluster, potentially resulting in overlapping clusters. Usually, intra-cluster relationships are not provided as part of the results. 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.

Some approaches completely avoid the high dimensionality problem by simply ordering the most used terms in the document and employing the first N terms [RES98], [RES99]. These strategies work well for single document

representation and for association of a limited number of documents, and even for some degree of clustering. However, they also lack a way to clearly relate different documents and display levels of similarity. Other approaches (such as the one by Carey et al. [CHR03]) combine a number of different strategies to allow various views of the same document set, potentially improving focusing and analysis tasks.

Maps resulting from the techniques mentioned above are meant to present various properties of documents, including content similarity, co-citation, term co-occurrence and attribute-value matchings, such as author or publication date. For a detailed description of available techniques for text document mapping and its applications, systems and challenges, we refer the reader to Borner et al. [BCB03].

Few systems for general multi-dimensional visualization add projections in their tool suite. The *Hybrid Information Visualization Environment* (HIVE) [RC03], however, implements clustering and layout algorithms to enable exploratory visualization of multidimensional data. An intuitive configuration process by the user is its main strength. Nevertheless, the layout algorithms implemented are approximations of the original FDP model, which fail to generate visualizations that group (separate) the data items based on their similarity (dissimilarity).

Projections have nevertheless been intensively explored in the context of visualizing unstructured data, particularly text document collections. Two typical layout approaches in this context are *InfoSky* [AKS*02b, GSAK04] and *Galaxies* [WTP*95, Wis99], the later one incorporated into the IN-SPIRETM system (see <http://in-spire.pnl.gov/>). Both systems display documents as points in 2D space following a similarity-based layout. Clustering and projections are employed to position groups of documents and to reduce the number of distance calculations. IN-SPIRE builds that into a dimensional reduction approach, similar to Single Value Decomposition or Latent Semantic Indexing [PRTV98, DFK*04]. InfoSky also creates a hierarchical display by embedding structure into the similarity relationships, so a user can focus in and out analogously to manipulating a telescope.

Within our research group, we have also developed — and made freely available — a tool suite for multidimensional visualization based on projections, called PEX (the Projection Explorer — <http://infoserver.lcad.icmc.usp.br/infovis2/PEX>). PEX is an experimental platform written in Java that generates visual maps of multi-dimensional data sets based on classical and novel projection and point placement strategies, with particular focus on text processing.

One way to measure the similarity between text documents in our case is by calculating cosine the distance [MPL06] between the vector representations. The other is using the “Kolmogorov distance” [TMP07]. This

measure is based on an approximation of the Kolmogorov complexity [LV97] using compression algorithms to compare one document against another. This can be done using the documents in their original form or, in some cases, in part of their original form. The result is not sensitive to tuning or to text dimensionality.

Visual representations of document maps assume various forms of 2D and 3D graphs and triangulations. Also popular is the landscape-type of display, generating heightfields over triangulations of the projected points. Landscape plots have been the choice of many useful presentations of documents before ([Wei01], [Wis99], [Cha93], [CC92]).

4.3. Visual text mining

The wide availability of information stored in the form of digital files has broadened our information universe to an amazing extent. The problem of extracting useful information from text collections is too often faced both by professionals and ordinary computer users, as they interact with, e.g., file systems, digital libraries, email communication, patent databases, news sites, and the Web in general. In this context, to extract information can mean both to look for a particular piece of knowledge, or to browse the collection in order to gain general knowledge about its contents and to infer relationships amongst documents that meet some user interest. Informed decisions many times depend on being able to explore this type of overwhelming information fast.

Users face difficulties in executing such exploratory search tasks, particularly when manipulating large text collections. Consider, for example, someone who just got a sheer amount of search results displayed as a ranked list. The user will likely inspect only a small subset of the whole list, and in doing so will perhaps miss connections between distinct pieces of information that might be of interest or relate to the query somehow. A ranked results list is generally adequate to meet specific information needs, but little encouraging if the user is in any sense trying to achieve a general view or a network of knowledge within a domain.

Help might come from text mining techniques [WIZD04] embedded into search and browsing engines. Semi-automatic mining algorithms can extract semantic patterns, such as similarity and associations from ordinary text documents or queries very well. Classification and clustering are typical text mining tasks. Their goal is to segment the corpus into groups of documents according to their similarity, where a criterion must be defined to compute similarity. Similar documents, as stated by this criterion, should be placed in the same group, dissimilar documents must be assigned to distinct groups. In classification, assignment of a document to a particular group is a supervised process in which the set of groups, or classes, is known beforehand. Clustering is a type of unsupervised classification, that is, the class distribution is not given and data set segmentation is based solely on

features extracted from documents. Partitioning into groups induces an organization that can help explore individual segments of the data set and obtain an overview of the collection from the characterization of the groups.

Classification and clustering have their drawbacks. Classification requires previous knowledge about the collection, which may not be available. For clustering, establishing similarity or dissimilarity amongst documents is not straightforward. Indeed, perception of similarity between text documents is sometimes subjective and depends on semantics and domain characteristics. This means that in some situations, a correct grouping would have assign certain 'border-line' documents to more than one group, and it may be that, after closer inspection, documents in the same group do not seem to be that well related to each other. So, grouping algorithms tend to hide these concepts, frequently very important in some exploration scenarios. Additionally, few techniques are capable of expressing additional relationships, intra-cluster or intra-class, leaving the user with 'pockets' of similar documents but no explanatory view of what is inside each pocket or what types of relationships exist within a group.

Other machine learning techniques, such as association rules extraction, can be used to extend support for user centered exploration of text data sets, since relationships within a document can be inferred based upon their sharing common terms. A term consists of one or multiple meaningful words. Given a set of terms, an association rule has the form $(t_1, t_2, t_3, \dots) \Leftarrow (t_i, t_{i+1}, \dots, t_n)$, meaning that for a document, the common occurrence of those terms on the right-hand side of the rule implies that all the terms on the left-hand side also occur. Two measures usually applied to express the importance of a particular rule are *support* and *confidence*. The support of a rule informs how frequently the terms on both sides of that rule appear together in the document set, and its confidence informs the percentage of the documents in the set that, having all right-hand side terms, also have the terms on the left-hand side. For instance, a rule expressed by $(beer, chocolate) \Leftarrow (peanuts, candy, aspirin)$ with a 30% support implies that all the five terms in the rule appear together in 30% of the whole collection. A confidence of 80% would mean that, from the collection, 80% of the documents that contain beer and chocolate also contain the other three terms in the rule. Mining of association rules has been applied to extract meaningful relationships between text documents based on their content [CNT04].

An association rule mining algorithm scans the data in search for possible associations that can be established for given support and confidence thresholds. The logical intuition in analyzing the rules so obtained is that documents that share term co-occurrences are likely to be addressing related issues, and therefore can be considered similar or correlated somehow. Nonetheless, the process easily generates a huge number of rules — the actual number being exponential in the number of terms. As a consequence, out of thousands of

association rules characteristically generated by a mining algorithm, maybe only a very small subset is of real interest to the user's task, impairing the exploration process. A filtering process is therefore necessary (and many times troublesome) to identify relevant rules.

In the pursuit of solutions capable of better matching user needs, the field of Visual Text Mining is gaining strength. It exploits the synergy between mining and abstract information visualization to create interactive visual representations of document collections for browsing and querying. The ability to couple visual representation with cues provided by mining algorithms is proving to be valuable to users trying to identify patterns with high semantical content, both globally and locally.

Chen [Che04] suggests that users build an internal cognitive map when navigating through a visual information space, analogous to real world navigation. In fact, many user-driven visual text mining approaches rely on so-called document maps — visual information spaces for user navigation that, similarly to geographical maps, spatially reflect one or more properties of the documents that may be of interest. A document map may be built from extracted information, such as co-citations or common citations, presence and distribution of topics, co-authorship, etc. It can also be based on text characterization such as content similarity. Maps can support a variety of exploratory tasks, connecting users with their own cognitive map while circumventing some of the inherent complexities of the underlying information space.

Displaying these maps in a form visually analog to familiar geographical maps has been pointed out as a major strength of map-based interfaces to document collections [Sku02]. Wise [Wis99] strongly argues in favor of an “ecological approach” to text visualization, in which visualizations are grounded in human perception capabilities — his sample visualizations are analogues to night sky and terrain models, whose interpretation is eased by capabilities wired into our brains as a result of our biological heritage. Several approaches exist for creating and displaying document maps. The underlying concept, however, is that proximity in the visual maps reflects some measure of document similarity.

5. Projection Based Visualization and its Application to Visual Text Mining

5.1. Projection techniques and point placement strategies

Multidimensional projection techniques can be split into two major groups, according to the functions f employed: *linear projection techniques*; and *nonlinear projection techniques*.

Linear projection techniques create linear combinations of the data attributes, defining them in a new orthogonal

basis of low dimension. A widely known linear technique is *Principal Component Analysis (PCA)* or *Karhunen-Loève Expansion* [Jol86]. PCA is a second-order technique, that is, it employs information embedded in the covariance matrix of the data. For m attributes, a covariance matrix is a $C_{m \times m}$ matrix whose element c_{ij} denotes the covariance between data attributes i and j . The covariance indicates the degree of linear relationship between the two attributes. Second-order techniques are particularly suitable for data presenting Gaussian (normal) distributions, since in this case it captures almost all data distribution.

The process adopted by PCA is to create the covariance matrix of the data, then decompose it into m eigenvectors with m eigenvalues. The first p eigenvectors with the largest eigenvalues are selected to transform the m -dimensional space into a p -dimensional space that retains the major variance of the data. The variance can be successfully captured even if $p \ll m$. Indeed, PCA is the technique that produces the best results in terms of the information loss. It retains as much as possible the relative distances among the data instances, whilst projecting them onto a low-dimensional space.

Although they perform well on Gaussian data, in handling data with nonlinear structures, such as clusters of arbitrary shapes or curved manifolds, linear techniques typically fail to capture the relevant patterns. In such cases, nonlinear techniques are better candidates. Rather than relying on linear combinations of the attributes, nonlinear techniques attempt to minimize a function of the information loss incurred in the projection. Normally, this function is based on the dissimilarities amongst the m -dimensional instances and on distances among the p -dimensional points. Hence, it does not require representing the original data as vectors, it is sufficient to have a mechanism to measure instance dissimilarity in the high-dimensional space.

Since nonlinear techniques perform an optimization process, their iterative nature is an additional advantage. Thereby, a user can observe the execution of the projection process and interrupt it if convenient. Another interesting feature is that adding new subsets of instances only requires a limited number of additional iterations. Linear techniques, on the other hand, demand the overall process to be entirely redone.

One example of a nonlinear projection technique is *Multidimensional Scaling (MDS)* [CC00]. Sprang from the psychophysics domain, MDS actually comprises a class of techniques aimed at mapping instances belonging to an m -dimensional space into instances on a p -dimensional space ($p \leq m$), striving to keep some distance relations. A well-known example of MDS technique is called *Sammon Mapping* [Sam64]. It starts by defining a function that indicates the amount of information loss incurred in the projection, and then applies an iterative nonlinear optimization method based on the gradient of such function to find a (local) min-

imum. This function is presented in Equation 1. One observes that it will reach a minimum when the dissimilarities $\delta(x_i, x_j)$ amongst the m -dimensional instances are close to the distances $d(f(x_i), f(x_j))$ between the p -dimensional points. Some normalization is also applied to favor the definition of more compact spaces for the projection.

$$S = \frac{1}{\sum_{i < j} \delta(x_i, x_j)} \sum_{i < j} \frac{(d(f(x_i), f(x_j)) - \delta(x_i, x_j))^2}{\delta(x_i, x_j)} \quad (1)$$

Amongst the various MDS techniques, the simplest ones are those based on *Force-Directed Placement (FDP)* [FR91, Cha96]. Originally proposed as a graph drawing heuristic, the FDP model aims at bringing a system composed of instances connected by imaginary springs into an equilibrium state. Initially, the instances are randomly placed in the system, and the forces generated by the springs are employed to iteratively push and pull the instances until reaching an equilibrium. In order to apply the FDP model as an MDS technique the spring forces must be proportional to the difference between the dissimilarity $\delta(x_i, x_j)$ among the m -dimensional instances, and the distances $d(f(x_i), f(x_j))$ among the p -dimensional points.

One example of the former strategy was presented in [TMN03], called *Force Scheme*. Different from the original idea of Eades, where each instance is moved once per iteration, Force moves each instance $n - 1$ times on an iteration. Thus, less iterations are necessary to bring the system to an equilibrium state. Although it reduces the model's complexity, each iteration is still $O(n^2)$. Aiming at reducing this complexity, Paulovich and Minghim [PM06] proposed a new method where the instances are first clustered, and the Force is applied considering the instances of each separated cluster, defining a model whose complexity is $O(n^{\frac{3}{2}})$. The core idea of ProjClus is to calculate the centroids of the initial clusters, then project these centroids onto the plane. Next, it separately projects each set of points defined by the clusters onto the same plane; then the technique assembles the final layout, positioning the clusters' projections according to the projection of their centroids. A related approach was applied by Andrews [KGM*01].

An example of a technique that also reduces the complexity of the iterations of FDP was presented by Chalmers [Cha96]. This approach reduces the complexity of an iteration using data samples in order to determine which instances are connected by the imaginary springs. Although this approach makes the iterations linear, the model complexity is still high due to the n iterations necessary to create the layout, being $O(n^2)$. Aiming at reducing this complexity, another approach was presented in [MRC02] (and extended in [MRC03]), which defines a FDP model with complexity $O(n^{\frac{3}{2}})$. In this approach a random sample S of \sqrt{n} instances is first projected using the Chalmers ap-

proach. Following, the remaining instances are interpolated from these instances. The process for the interpolation is the one that makes this technique $O(n^{\frac{3}{2}})$, therefore, Morrison et al. [MC04] suggest a modification of this interpolation reducing the final complexity to $O(n^{\frac{5}{4}})$, and Jourdan and Melancon [JM04] suggest a further approach to reduce it to $O(n \log n)$.

Besides ProjClus, we have developed two other distinct nonlinear projection techniques, all illustrated before in Section 2: *Least-Square Projection (LSP)* [PNML06] and a point placement strategy that builds a similarity tree, called N-J (*neighbor-joining*) tree map [CPMT07].

The LSP technique is a generalization of an approach for mesh-recovering and mesh-editing in order to deal with high dimensional spaces. In this technique, a subset of m -dimensional points are projected onto the plane, and the remaining points are projected using an interpolation strategy that considers only the neighborhood of the m -dimensional points.

ProjClus and LSP are high-precision and fast projection techniques, which are suitable to handle points belonging to nonlinear sparse spaces — such as the one generated from document vector representations (see Section 3.1.15) — since they take into account neighborhoods of the original points.

One approach for point placement that differs from the conventional view of grouping by proximity on the 2D plane, is the creation and drawing of similarity trees that reflect the similarity relationship calculated by the measure employed [CPMT07]. To do that, the technique uses a phylogeny reconstruction algorithm, called neighbor-joining. Points that are closer together in the whole set are joined in a tree and replaced in the list by a 'combined' node which then enters in the search process together with the remaining points in the data set. The N-J tree is capable of representing relationships that allows the user to quickly recover information detected by the similarity metric.

5.2. Mapping text collections via projections and point placement

The result is a planar mapping of a point set X representing the documents (we call this 2D placement a *map*), which can be used to explore the document collection represented by X . As is going to be illustrated here, the techniques we have developed are capable of mapping documents in such a way that text data dealing with common subjects form groups that are visually apart from other groups, thus allowing identification of patterns in the text set. An example of such a map is given in Figure 11. It shows a document map of a collection of 574 scientific papers belonging to three different subjects, previously manually classified. In the pictures, different classes have different colors (red is Case-Based Reasoning (CBR), blue is Information Retrieval

(IR), and green is Inductive Logic Programming (ILP)). The small greenish group in the rectangle shows the placement of five papers on sonification (the use of sound to display information). That particular group refers to the previously-mentioned five papers reflecting the evolution of our sonification system, which are strongly correlated. Figure 11(a) shows that they are adequately placed in the same neighborhood. Figure 11(b) demonstrates that, when mapped in conjunction with the remaining papers, this group was placed in the outer boundary of the map and nearby papers in the data set whose main subject was audio retrieval (they are the points with colors other than red in Figure 11(c)). Another aspect that can be noted in the map is the fact that the CBR and ILP groups are more compact, whereas the IR set is spread out over the entire map, a visual analogy to the characteristics of the subject areas, the first two being more mature and therefore having a set of their own nomenclature and techniques, while IR relies on resources from various sources including those of ILP and CBR.

5.3. Topic extraction and visualization

5.3.1. Topic identification by covariance

Although a projection can help locating groups of related documents, the initial map does not reveal information within these groups. In order to extract information about these groups, a group can be selected and a label can be generated that aims at identifying the main topics discussed within that group.

From a group of selected documents, a label is constructed by an algorithm that first chooses the pair of words with the highest covariance in the vector representation of the group. Then, for each remaining (non-selected) word it computes the mean of the covariances relative to these first two words. If the covariance of two terms approaches zero, it means that the terms are independent. If this is a significant value relative to the highest covariance, e.g., above a defined percentage threshold (in this paper we used 50%) the word is added to the label. This is the way the labeling by term covariance is implemented in PEx, and that was used in the illustrations of Section 2. In those pictures, the first two terms on the left side of a label are the ones that present the greatest covariance, and are indicative of the main topic in the group.

The term covariance is calculated according to Equation 6.

$$\text{cov}(t_i, t_j) = \frac{1}{n-1} \sum_{k=1}^n (t_{ki} - \bar{t}_i)(t_{kj} - \bar{t}_j) \quad (2)$$

where \bar{t}_i is the mean of the i^{th} term t_i , and t_{ki} and t_{kj} are the values of the i^{th} and j^{th} terms for the k^{th} document.

In order to illustrate the representativeness of the term covariance further, the examples of Figures 11 and 12 have

defined the labeling in a slightly different manner than the one illustrated before, and currently implemented in PEx. Figure 12 shows a document map of the same data set of Figure 11, showing the resulting labels after group selection by the user.

The first two terms on the left side of a label are the ones that present the greatest covariance. The third one on the right side is the term which has the greatest *mean covariance* according to the labels already chosen. The mean covariance is calculated as the mean between the covariances of a term taking two other terms. The numbers between parenthesis are the covariances.

If the covariance approaches zero, it means that the terms are independent. On the label, this number can indicate if the three terms are equally related or if only the first two are. If the number on the left side is much greater than the number on the right side, it means that the third term is not deeply related to the other two. But if these numbers are close, it means that the three terms occur frequently together.

For instance, in Figure 12(b), considering the label *[learning,logic](60.44) >> [programming](44.07)* and dividing the right covariance by the left we get $\frac{44.07}{60.44} \approx 0.73$. Calculating the same rate for the label *[information,retrieval](101.76) >> [text](26.56)*, we get $\frac{26.56}{101.76} \approx 0.26$. Thus, it is possible to infer that the terms “learning” and “logic” are much more related with “programming” than the terms “information” and “retrieval” are with “text”.

Notice that the covariance is not a causality relation between terms, that is, it does not indicate that one term occurs due to the other one. It is only a measure of the degree to which two terms vary together. For example, on the map of Figure 12(b), which is a view of part of the Figure 12(a), the terms “secret” and “sharing” present a high degree of covariance, but it is not possible to establish that the term “secret” will cause the term “sharing” or the other way round. Another important aspect is that using two terms only, it is not normally possible to direct infer why they are related. This is the point where the third term (as well as the others) of the label can help. The third term for the terms “secret” and “sharing” is “schemes”, thus it is possible to infer that the main topic of the document used to create such label is somehow related to cryptography.

Covariance is useful to identify main topics in a group of documents. A more advanced technique is necessary to obtain more detailed automatic topic extraction, exploring further the sub-topics in a group as well as additional topics of documents that may not match the main topics extracted by covariance. The next section presents one such technique, which handles detailed and automatic topic extraction using association rules. For a more detailed view of that subject, we add a previously written technical report in Appendix B. We also refer to [LPMP07] and [PLOM07].

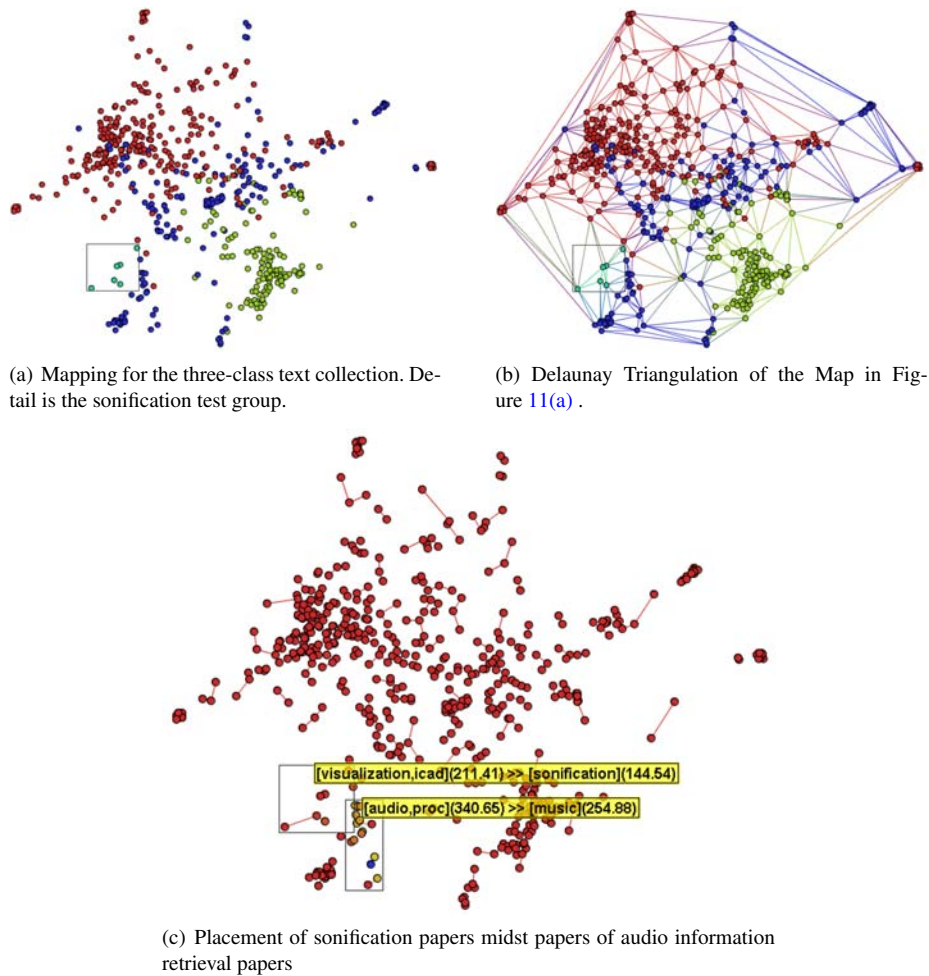


Figure 11: Document map of a collection of scientific papers belonging to three different main subjects, manually classified, plus a test group (inside the rectangle).

5.3.2. Topic extraction by sequential covering induction of association rules

An appropriate set of association rules derived from a collection of text documents can be used to describe a context in which a term appears or also the context or topic related to a subset of documents. When mining association rules from text, an association rule (AR) is an implication of the form $X \Rightarrow Y$, where $X \cap Y = \emptyset$ and they are both subsets of L , where $L = l_1, l_2, \dots, l_m$ is a set of literals or items (each item representing a term from the bag of words). A transaction T is a set of items $T \subseteq L$ that represents a document from the corpus C . The rule $X \Rightarrow Y$ holds in the document set C with confidence c if $c\%$ of the documents in C that contain X also contain Y . The rule $X \Rightarrow Y$ has support $s\%$ in C , if $s\%$ of documents in C contain $X \cup Y$.

A rule $R_i: X \Rightarrow Y$ with significant support in C means that

a set of terms $l_i \in X \cup Y$ are found together in a large subset of documents from C . Moreover, if R_i has high confidence, then the occurrence of X (the body) means high probability of the occurrence of Y (the head), at least in C .

In [LPMP07], an algorithm is presented to produce and select good association rules in order to describe the main subject or topic in a selected set of documents S_k . That algorithm deals with the problem of the large number of association rules as follows. Instead of post-pruning the set of rules generated from all documents in S_k by some rule quality measure, it generates rules that contain at least one term (seed) present in a selected set of most relevant terms. The term relevance is given by a weight that favors terms with higher frequency in the selection than in the rest of the corpus. An arbitrary number of rules, usually 1– 3, with the highest term weight summation is then selected for display as label for the group selected. Eventually, some documents

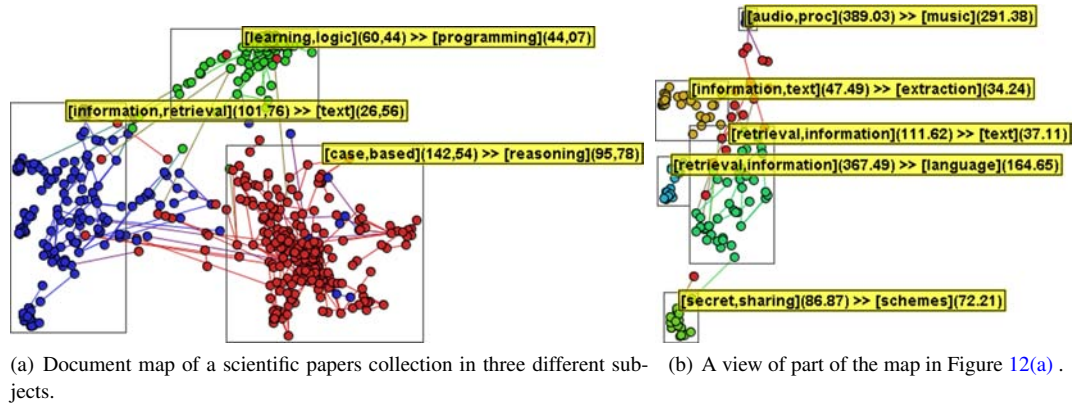


Figure 12: Example of labels for document maps.

carry topics that are not described by these selected rules, that is, they are not covered by the rule.

Another algorithm uses a sequential covering strategy in order to extract topic-targeted rules from the documents uncovered by the main rule. Additionally, instead of just selecting the n highest weighted rules, it keeps only those rules that provide additional covering over the previous ones. This reduces the amount of redundancy found in the selected rules, considering that if two rules cover the same subset of documents, only the one with the highest weight is kept.

The Algorithm has two nested loops. In the inner loop, the iterative algorithm for generating and ranking rules over a selected area S_k of the projection leads to a set of rules SR . This process is repeated (the outer loop) removing covered documents during the iteration until no document is left or the rule generation process outputs an empty set of rules. The function coverage (AR_i, U) of Algorithm 1 denotes the number of documents from U that support the rule AR_i .

Even considering that each selected rule must cover documents not covered by any other previously selected rule, a considerable number of rules may be generated for a given selection of documents. To reduce cluttering, we apply a rule grouping strategy that relies both on literals in the rule and on overlap of covered documents. Intuitively, if rules share some of their literals and cover roughly the same set of documents, they are likely to represent the same topic, but described in a slightly different way. Thus, instead of presenting a label for each rule, only one label per group of rules is displayed. A rule AR_i joins a group of rules G_j , if (i) AR_i has at least one term $t \in T_{G_j}$, where T_{G_j} is the set of literals found in rules already in the group G_j , and (ii) there is an overlap between the set of documents covered by the rule, and the set of documents that are covered by any of the rules in the group. This document coverage overlap should be equal to or higher than a given constant α . Here, we used $\alpha = 0.5$. To compute the overlap, we use:

$$overlap = \frac{|\cap D_{AR_i}, D_{G_j}|}{\min(|D_{AR_i}|, |D_{G_j}|)}$$

where D_{AR_i} is the set of documents covered by the rule AR_i , and

$$D_{G_j} = \bigcup_{AR_i \in G_j} D_{AR_i}$$

If a rule does not join any group, a new group is formed with that rule as an initial element. The label of a group of rules begins with the highest weighting term taken from the rule with the highest support in the group. The label shows also this high support rule followed by all the terms that are found in the other rules of the group and that have not yet appeared. For instance, the label “mosque:[suicide]←[mosque], (27/20%;90%):iraq:baghdad:” is given to a rule group where the rule “[suicide]←[mosque]” has the highest support among the rules in that group (27 documents or 20% of selected points S_k), “mosque” is its term with the highest weight, and 90% is its confidence. In the example, “iraq” and “baghdad” are the terms found in other rules from the group that does not appear in the highest support rule.

Figure 25 demonstrates the potential of this technique in aiding exploration of document collections. To build the map, the flash news data set used in Section 2 was employed. The picture presents the whole set of documents colored according to a clustering algorithm in PEx. Labels have been generated for each of the clusters and a few of them were selected for display.

Labels such as those, which reflect topics in the data set, can be summarized by displaying the main rule and a list of the remaining terms, ordering them according to rule coverage and term relevance within the rules. This prevents too

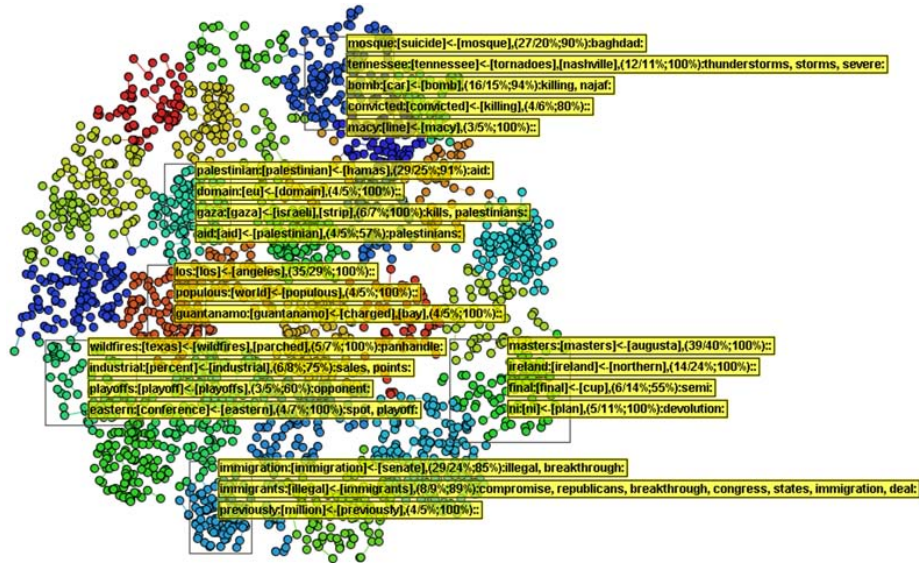


Figure 13: Document map of the flash news data set. Projection by ProjClus. Individual colors denote clusters of documents. Shown labels were generated from groups of rules induced using some of the document groups.

much cluttering of the image, although it hides the various degrees of abstraction revealed by the rules themselves.

5.4. Further Examples

See Slides attached.

6. Conclusions, Current Challenges, Future Trends

The field of mapping documents for content analysis, particularly in the light of new trends in visual and text analytics, is extremely challenging and raises many complex issues.

The first one relates to the extremely different applications that rely on text and, within them, the varying tasks associated with the exploration of such maps. There are applications as diverse as Web search information gathering and analysis of medical information from patient records. As illustrated by some of the examples in these tutorial notes, not only text can be transformed into table data, but also table data can be transformed into textual information. The matter of customization is an important one and there are many parts of the various techniques for text mapping that can be adjusted to fit particular data sets and particular tasks.

The problem of customization also bears relationship to one of the critical challenges of these applications: the pre-processing stage of the mining and visualization processes. Stopwords definition is many times user controlled and is very rarely trivial. Yet, it is a necessary step in most applications of mining, visualization and visual mining. In the

case of applications that involve safety of high human involvement (such as medicine, forensics, detection of illegal material), particular depth is due in the definition of vocabulary and in linguistic processing before conclusions can be reached.

The issue of what is important to define as similar in a particular setup of application and task also points to the development of new approaches for calculation of similarities between pairs of documents. Proper strategies can be developed that take into account the homogeneity and semantic levels of a text collection as well as the context in which it is being analyzed. The results of employing Association Rules for topics thus far indicates that there are levels of abstraction that can be extracted employing similar approaches with the target of finding an appropriate combination of concepts or a relevant vocabulary for exploration tasks.

The issue of quality of projections is still open. The classical stress parameter, targeted for optimization strategies for point placement, is of little help to reflect effectiveness of grouping and separation of documents by their content. Additional paradigms and procedures need to be developed.

Closer integration of visual aids with mining is also expected to be one of the hottest issues to be addressed in the coming years. Also are the coordination and multiple views between different visual representations, and even between different mining algorithms. Information gathered from the visual process can be fed into algorithms such as classification. Conversely, mining results can be reflected back onto the visualizations, either by changing its visual (and why not

aural and tactile) attributes, or by feeding into the parameter adjustment of the visualization pipeline [Hei07].

Naturally, a crucial issue regarding textual data sets is scalability. While a few techniques can handle large numbers of documents, there is a compromise in the quality of the resulting map regarding content associations. The current state of this technology seems to point towards two types of analysis tasks: one for massive quantities of text, which relies on efficient algorithms (e.g., visually supported or based on mining and intelligent search strategies) to reduce the data set; and a second type that takes a few thousand individual documents to be visually explored in an integrated mining and visualization environment such as that presented here.

Finally, two issues of great importance, touched upon only superficially in this material due to space constraints, are the interaction paradigms and the evolution in time of a particular data set. New visual approaches imply different strategies for exploration, particularly when coupled with processing by mining algorithms. The issue of time is also extremely pertinent in many applications of textual exploration (e.g., medicine, news, research topics evolution, etc.). They need to be addressed through proper models, and, in the case of the exploration techniques, inspired by user evaluations of current available technology.

7. Acknowledgments

This work has been mostly supported by the Brazilian research support agencies FAPESP (grants 03/02815-0, 04/01756-2, 04/09888-5 and 04/07866-4) and CNPq (grant 304758/2005-1). Haim Levkowitz's research was partially conducted while he was a Fulbright US Scholar to Brazil, August 2004 – January 2005.

The authors wish to acknowledge the following profound contributions to the results behind the research reflected here:

- Fernando Vieira Paulovich, PhD student
- Roberto Pinho, PhD student
- Guilherme Pimentel Telles, research partner, ICMC
- Alneu de Andrade Lopes, research partner, ICMC
- Maria Cristina Ferreira de Oliveira, research partner, ICMC
- Luis Gustavo Nonato, research partner, ICMC
- Lionis Watanabe, Wagner Franchin and Ana Maria Cuadros Valdivia, MsC students
- Pedro Vilela and Renato Rodrigues, undergrad students

Rosane Minghim also acknowledges the enthusiasm and productive research exchange currently in place with Dr. Anton Heijs, Trepapel Inc., The Netherlands, and Charl Botha, Technological University of Delft, The Netherlands.

Appendices

A. Technical Report: VISUAL MAPPING OF TEXT COLLECTIONS USING AN APPROXIMATION OF THE KOLMOGOROV COMPLEXITY

Guilherme P. Telles, Rosane Minghim and Fernando V. Paulovich

{gpt,rminghim,paulovic}@icmc.usp.br
ICMC - University of São Paulo, Brazil

The generation of content-based text maps is an important issue to support exploration of information and to help find relevant reading material in increasingly complex document databases. Most techniques that help relate or visualize texts rely on a vector representation that is, at its best, ad-hoc as to its parametrization. This paper presents a novel approach capable of generating a map of documents without the painstaking pre-processing steps, by comparing text against text through an approximation of the Kolmogorov complexity. The similarity measure taken from that analysis is then used to map data in 2D by applying fast multidimensional projection techniques (instead of dimensionality reduction or random initial point placement). The resulting maps show a high degree of content separation and good grouping of similar documents. The approach can be used to map text collections in a variety of applications and the map can be interacted with to further explore text groups. By avoiding vector representation our technique decreases the bias characteristic of that approach and the need for user knowledge of the process. The approach also lends itself to incremental processing for reduction of computational costs.

A.1. Introduction

In a set up of document collections, such as text databases or Internet search results, it is not easy to locate reading material even with an effective ranking system. In the applications targeted here the goal of a user's analysis is to locate relevant documents to examine or study amongst a considerable number of recovered texts. We approach that by studying effective mapping strategies that are capable of coding relationships amongst documents geometrically and visually. An interactive map based on document content can be explored to locate a text relevant to a query or to another target text and to find groups of similar or related documents. This is usually done after a pre-filtering process that has narrowed the collection down to a range of few hundred to a couple of thousand texts. Typical applications are research, education and training.

Most text mapping strategies are based on clustering or dimensional reduction that rely on text vector space representations whereby, after a considerably lengthy pre-processing step, texts in a collection are represented as a vector of many dimensions. Each dimension is a relevant term in the text

set. One of the problems with this largely used vector approach is that the dimension of the final data set can reach the thousands easily. It is a well know fact that, as the dimension gets larger, the ability to properly infer vector distances gets impaired, prompting the need for some sort of attribute selection.

As far as processing speed is concerned, the transformation of that representation into a map of some sort may involve either dimensional reduction or clustering techniques, which can be rather slow themselves, or on faster projections, without significant quality loss ([MPL06, PNML06]). The fact remains, though, that the pre-processing step involved in all of them include various procedures, such as stopwords elimination and stemming, that can be affected by various parameter settings. Other adjustments for feature selection and frequency analysis are often necessary. These adjustments have large influence on the outcome, sometimes prompting more than a few iterations before the result is satisfactory, and making the process too sensitive to change in the subject target texts. In most real cases, therefore, it is not possible to tell in advance what the right tuning for the pre-processing is.

This paper proposes a novel approach to text mapping based on direct comparison between texts contents without the need for any preprocessing, by overriding the vector representation altogether. We calculate an approximation of Kolmogorov complexity ([LV97]) as a similarity measure between texts. That is used as a distance value to generate multi-dimensional projections into 2D space by means of a fast approach [PNML06]. The resulting maps can be interacted with in a form that allows further exploration.

The following section offers a review of relevant text mapping literature and the role of projections in its context. Section A.3 describes the multi-dimensional projection procedure employed here step by step. A summary of the theory and implementation of the Kolmogorov 'distance' is given in Section A.4. Results and comparison to other maps are given in Section A.5, which is followed by analysis, conclusion and further work discussion.

A.2. Previous Work

Due to the complexity and variety of the information and scenarios involved in text examination, alternative means of mappings text sets must be sought. Here we review the works in the literature that deal with this problem that, in our view, cover the main issues relating to visual mapping techniques for documents.

A number of different techniques for visualization of textual results from Web and other searches have being deployed ([ABY03], [LA00], [SCL*99], [BY96], [Cha96]). While these techniques are capable of displaying large text bodies, they tend to make location of relevant reading material more troublesome. Our focus in this work is to provide

complementary tools to support mapping of documents in a way that helps locate neighboring similarities between texts and groups of texts. So we assume a pre-filtering task that reduces the universe of targeted documents to a few hundreds (maybe thousands) of texts in a few areas of interest (not necessarily pre-determined).

Many techniques for text visualization exist that search for a representation of the content of an individual text (e.g. [MWBF98], [RES99, RES98]), of text collections (e.g. [LA00], [BCG*99], [Wei01]), or of themes approached in texts (e.g. [HHWN02], [Wis99], [WTP*95]) in order to meet the above mentioned targets.

Usually text processing tasks employ the vector space model [SB88, Sal91] whereby texts are represented as points in a vector space. In this representation each text is a vector with dimensions represented by terms (n-grams). The vector coordinates are the weights of the terms based on their frequency. Typically, dimensions reach the thousands even for small to medium databases. Transformation of a text collection into a vector space is preceded by elimination of non-influential words (such as stopwords), reduction of words to their radicals (stemming), and frequency counting of some sort (various exist). The initial representation is followed by reduction in space dimensions, typically involving cutting off words that are too frequent or too rare in that particular collection, and clustering dimensions to generate new 'combined' attributes, in an attempt to overcome the dimensionality curse.

The most common way to extract structure from a text collection is by applying some sort of dimensional reduction technique over the resulting vector representation. This is the case of systems based on Multi-dimensional Scaling (MDS), Principal Component Analysis (PCA) or Latent Semantic Analysis (LSA), that work with statistical measures for subspace reduction, and Self-Organizing Maps (SOM), that employ neural computation ([Wei01], [Wis99], [BCG*99], [KHLK98], [WTP*95]). Those techniques can be used to plot the original data in bi-dimensional (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 [HWR03]: 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 an 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 work by Lopes et. al [LMMP06] LSA has been successfully ap-

plied for the generation of document maps with high local content relationships, but the high computational cost remains a problem as well as the handling of vector transformations.

Another recurring strategy for dealing with the organization of information from a text collection is document clustering ([CHR03], [RK04]), many times employed in combination with dimensional reduction and SOM ([IR01], [Wei01], [LA00]). They provide a way of relating documents with varying success rates. When clustering techniques are applied, here too the intra-cluster relations are 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.

Point placement strategies and force-based point placement improvement have been used before to generate document displays [Cha96, GSAK04, AKS*02a]. Although still based on vector representation, they can avoid partly or completely the extensive calculations needed in dimensional reduction techniques by starting with a semi-random point placement and re-adjusting their position based on attraction by similarity.

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 [RES98]. 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 a way of clearly relating different documents and displaying levels of similarity. Other approaches (such as the one by Carey and others [CHR03]) combine a number of different strategies to allow various views of the same document set, potentially improving focusing and analysis tasks.

Final maps resulting from the techniques mentioned above are meant to analyze a number of properties of documents, including similarity, co-citation, term co-occurrence and various others. We refer to the work of Katy Borner and others [BCB03] for a detailed description of the previously available techniques for text mapping and its applications, systems and challenges. A few systems are being developed dedicated to viewing maps from multi-dimensional data and some of them are particularly dedicated to text collections. One recently published system [GSAK04] adds representational power to the conventional ways of plotting text as points in 2D by separating their contents in thematic areas and handling levels of interaction by hierarchical organization.

In general the methods discussed above lack the ability to determine levels of associations between texts contents. Others are computationally expensive. Faster mapping approaches with the ability of associating texts by similarity have been put forward [MPL06, PNML06, PM06]. Their gain in processing time is attained by using projection techniques,

which are faster compared to dimension reduction and also provide an initial point placement prone to speed up force-based improvement schemes. But those too are based on the vector representations.

Text vector representations, although very useful and largely employed, many times are cause for concern. They tend to impose bias and be difficult to tune. Depending on the choice of pre-processing parametrizations (such as Luhn's cut, vocabulary, types of stemming, types of frequency count, type of feature clustering or selection), the outcome of the analysis and displays of text collections can be highly affected. That also makes it difficult for lay users to employ the representation inside its usual context without knowledge of the processing, pre-processing and visualization techniques.

Another issue that impairs general use of vector-based techniques is its adaptability to incremental processing. Adding new texts to a previously existing collection mostly implies in rebuilding the visualization (including pre-processing) almost from start, propagating possible limitations to every map formation.

The technique presented here is based on projection techniques that have been proven useful to group and separate data when the similarity measure is sufficiently powerful [TMN03, MPL06]. To be able to function, these projections need only the distance between data points (in our case, texts), and not the original data themselves. The contribution of our technique is in producing such a distance between texts through a similarity measure that completely avoids the conventional vector representation. This way it eliminates the need for the ad-hoc pre-processing steps necessary to build that representation and avoids the problem of treating high dimensionality, a real trouble for texts. This measure is based on an approximation of the Kolmogorov complexity [LV97] and is calculated by comparing text against text. This can be done using the texts in their original form or, in some cases, in part of their original form. The results have reproduced separation and grouping advantages of previous methods and is not sensitive to tuning or to text dimensionality. These features also make the method easier to use. Opposed to most vector based mappings, it is naturally adaptable to incremental processing, with affordable storage overhead.

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. Additionally, surfaces are highly interactive and familiar to most users. Landscape plots have been the choice of many useful presentations of texts before ([Wei01], [Wis99], [Cha93], [CC92]). This feature combined with the absence of pre-processing allows interpretation of mappings even by users with little expertise in the field. The surface representation of our technique is enriched by mapping further significant information to visual attributes (such

as lines, colors and height) and aural attributes (such as pitch and timbre). The final map can be explored to the advantage of users interested in having an overview of a set of texts, locating important texts in corpora, or finding useful associations between texts, thus selecting material to read or study.

A.3. Projection techniques for text visualization

Methods of data projection into lower dimensions have the advantage over dimensionality reduction that they are much faster and, depending on the type of projection, good for incremental processing. A previous work [TMN03] has shown the advantages of projection techniques based on distance metrics to obtain useful views of multi-dimensional data sets. When the distance calculation captures significant data set features, they are capable of separating the data collection into groups and result in good association between neighboring individuals. Additionally, those techniques lend themselves to landscape plots. Their application for mapping texts represented in the vector space analogy was tested before with satisfactory results [MPL06]. In this work the same types of projections are used to map points into a plane except the distance between texts is now calculated without generating the vector representation (see Section A.4).

Different from other techniques that can be used to map data into 2D or 3D, such as dimensional reduction, clustering, or point placement strategies that start from a random or semi-random 2D display, the goal of distance-based projection techniques – e.g. Fastmap [FL95] and NNP [TMN03] – 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 bi-dimensional 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 . Find 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 minimum for every pair of points $x_i, x_j \in X$.

The set P is called a projection. In this type of projection, it is of great importance the definition of a proper proximity criterion, calculated in our case from the Kolmogorov complexity estimation.

We have used two projection algorithms in our experiments, with similar results: Fastmap and NNP. The first realizes hyperplane projection and the second performs geometric placement in neighborhoods. Each projection is improved by a projection improvement scheme called Force [TMN03], that enhances those projections by recovering part of the information lost during the mapping process, in a similar procedure as that adopted before by other researchers, such as

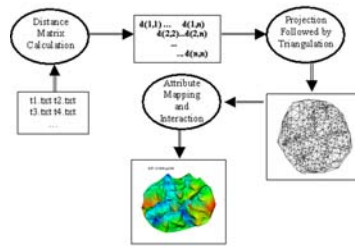


Figure 14: The whole mapping process. No pre-processing actually needed.

Chalmers [Cha96]. What the Force scheme does is iteratively approach points projected too far and repel points that were projected closer than they should have. It does that by a fraction of the ideal distance at each iteration. The computational cost of such improvement is minimum due to the initial point positioning by projection. The Force scheme also provides an overall measurement of the projection error (see [TMN03]).

The resulting points, now in 2D space, are connected by triangulation, thus to produce a surface, allowing interaction for exploration of text content. On top of that surface, text properties can be mapped to display attributes to support exploration. In our case, color, height and sound were employed to complete the map. Each attribute (color, height or sound) can map text clustering, or text category (in case it is pre-determined), as well as additional information such as year of publication, number of citations, rank, and so on.

The complete set of steps taken to build a map based on projection from Kolmogorov distance (or any other distance measure for that matter) is the following:

1. calculation of a triangular distance matrix comparing all texts and judging their similarity;
2. projection the points (texts) onto bidimensional space using a fast algorithm, followed by an improvement strategy;
3. triangulation of the data.

Figure 14 illustrates the complete process, and shows one possible resulting map. On top of the map, color (as well as height) were used to show clustering of the projected texts.

The calculation of the distance between texts is presented in the next section.

A.4. Kolmogorov Complexity as a means to define distance between texts

Intuitively, the Kolmogorov complexity is a measure of the amount of information that a message contains. It can also be seen as a measure of randomness of a string or as the length of a string that results after perfect compression. An extensive treatment on the Kolmogorov complexity appears

in Li and Vitányi's book [LV97]. A text can be seen as a string, so the discussion that follows applies to texts directly.

Formally, the Kolmogorov complexity of a string y , $K(y)$, is the size of the smallest algorithm that outputs y . Any formal notion of algorithm can be applied, such as Turing machines. The conditional Kolmogorov complexity of a string y given a string x , $K(y|x)$ is the size of the smallest algorithm that outputs y when x is given as input. Intuitively the conditional Kolmogorov complexity is the amount of information in y that is not known by x . The Kolmogorov complexity considered here is the prefix version, where algorithms are considered to be prefix-free, that is, no algorithm is a proper prefix of another.

The Kolmogorov complexity is not computable but it can be approximated using compression. Let xy denote the concatenation of strings x and y . Li and coworkers [LBC*01] defined the normalized distance between x and y

$$d(x,y) = 1 - \frac{K(x) - K(x|y)}{K(xy)} \quad (3)$$

and showed that $d(x,y)$ is a metric up to logarithmic additive terms.

Let $C(x)$ denote the length of the compressed version of a string x . Cilibrasi and Vitányi have shown that the normalized compression distance

$$NCD(x,y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}} \quad (4)$$

is a quasi-universal metric when the compressor in use is normal.

Kolmogorov complexity have been applied theoretical computer science [PSS81], clustering [CV05], plagiarism detection [CLMS02], phylogeny [LBC*01] and others [CV05].

In this work we used the LZW compression algorithm [Wel84] to evaluate an approximation of $d(x,y)$ denoted $d'(x,y)$:

$$d'(x,y) = 1 - \frac{C(y) - C(y|x)}{C(xy)} \quad (5)$$

We have evaluated the terms $C(y)$, $C(y|x)$ and $C(xy)$ as follows. The term $C(y)$ is the number of table positions that LZW would output while compressing y minus the number of table positions that LZW would output when the input is a string of $|y|$ equal symbols plus one. The subtraction is a tentative to overcome the limitations imposed by the very nature of LZW. The term $C(xy)$ is evaluated in the same way. The term $C(y|x)$ is the number of table positions that LZW

would output while compressing y , without counting the positions that belong to the compression table for x . That is, a table position is counted only if it does not belong to the table constructed during the compression of x . Our algorithm was implemented in Perl, using a hash for the table.

In our tests, we have also used the CompLearn [Paga] package together with gzip [Pagb] (chosen for speed) to evaluate NCD. We have noticed slightly better results using $d'(x, y)$ (see Section A.5), although with poorer performance.

Generating a distance table for n texts t_1, t_2, \dots, t_n with lengths l_1, l_2, \dots, l_n requires computing $C(t_i)$ for every text, and $C(t_i|t_j)$ and $C(t_i t_j)$ for every pair such that $i \neq j$. Then the cost of the distance table construction is $O(s^2)$ on the average, where $s = \sum_{i=1}^n l_i$ and with average cost $O(1)$ for a hash operation.

Adding a new text t_{n+1} to the set does not require recomputing the whole matrix. If we store the values of $C(t_i)$ and the LZW table produced for every t_i , it is enough to calculate $C(t_i|t_{n+1})$ and $C(t_i t_{n+1})$ for $1 \leq i \leq n$, at cost $O(s + l_{i+1})$ on the average. The length of LZW table for a text of length l is $O(l)$, so storing them requires an affordable amount of space.

A.5. Results

In the remaining of this text we shall refer to the similarity calculation based on NCD (Equation 4) as NCD and we shall refer to our similarity calculation based on compression by LZW (Equation 5) as k-lzw. In general, we refer to distance calculation based on Kolmogorov complexity approximation Kolmogorov distances or k-distances.

One of the goals of our maps is being able to distribute the files onto a surface automatically, allowing proximity of closely related contents, association between documents and groups of documents, and display of additional information related to them on the same map. By doing projection based on distance metrics it has been possible to obtain results that match those goals, provided the distance measure is capable of coding well content relationships.

To evaluate the ability of a particular distance measure to map corpus content, we have used text collections previously classified as belonging to general areas of knowledge. This classification is based purely on the source of the papers, and therefore some degree of overlapping is expected due to the presence of similar concepts or techniques cross-areas.

Pseudo-classes on the maps were colored to reflect mapping results visually, showing how those documents of the same class were distributed over the surface. Figure 15 shows the result of one final map from k-lzw for a corpus made out of papers from three basic areas: Inductive Logic Programming (ILP), Case-based Reasoning (CBR) and Information Retrieval (IR). It shows that this mapping is capa-

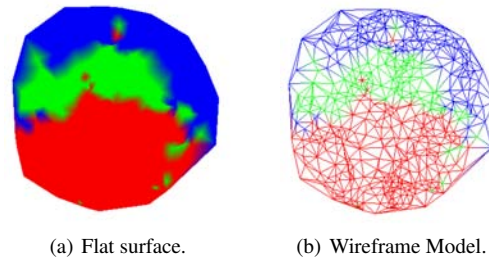


Figure 15: Mapping of scientific documents related to three different primary sources (CBR is red; ILP is green; IR is blue.).

ble of keeping most of the documents of the same class in the same region of the map.

Processing times for calculation of k-distances for a whole data set are quite high. The set in Figure 15, comprising 574 files took 2h30min to process from scratch in a Pentium IV processor of 3GHz. Opposite to conventional techniques based on frequency count, though, this type of processing is incremental and adding new documents does not require recalculation of previously obtained results. Only the new distances must be processed as databases increase in size, as mentioned in Section A.4.

Table 1 gives details of the documents used to process the remaining tests. The first scientific papers data set (corpus1) included title, authors, abstract and references from a number of texts. CBR and ILP subsets were taken from journals on those subjects. The IR and SON (sonification) subsets were articles obtained as a result of Internet searches pre-filtered to comply to those pseudo-classes. Those were all collected by members of our team. The corpus2 set was recovered from an Internet repository and comprehends files in the ISI format on the subjects of Bibliographic Coupling (BC), Cocitation Analysis (SC), Milgrams (MG) and Information Visualization (IV)(from ella.slis.indiana.edu/~katy/outgoing/hitcite/{bc,sc,mb,iv}.txt). The remaining sets are messages from news discussion groups recovered from an internet repository (see Hettich, S. and Bay, S. D. (1999). The UCI KDD Archive [<http://kdd.ics.uci.edu>]. Irvine, CA: University of California, Department of Information and Computer Science).

Previous results have shown that distance based projections of text collections based on vector representation with conventional distance metrics (such as cosine) can provide a point placement with good separation between general subject areas as well as good grouping of similar documents. This process, that entails vector representation possibly followed by k-means attribute combination, followed by projection with improvement scheme, has been called IDMAP (Interactive Document Map) [MPL06]. We call the mapping realized here (projection with improvement scheme based on

Table 1: Datasets used in the tests

| Set | Areas | General Content | Files |
|----------|-----------------------|----------------------|-------|
| corpus1 | CBR+IR+ILP+SON | Scientific Documents | 675 |
| corpus2 | SC+BC+MG+IV | ISI Files | 1624 |
| message1 | atheism+graphics | discussion | 200 |
| message2 | atheism+graphics | group messages | 300 |
| message3 | +baseball | discussion | 700 |
| message4 | seven varied subjects | group messages | 1000 |
| message4 | ten varied subjects | group messages | 1000 |

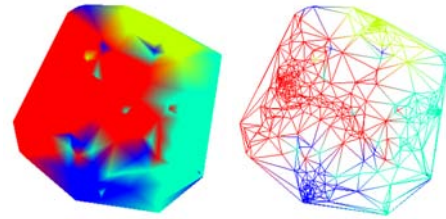
k-distances) Kolmomap. We compare these two approaches for visual result.

Figures 16 and 17 show the visual results of some of the data sets in both IDMAP and Kolmomap. It can be seen that the result is comparable to that obtained with IDMAP in terms of region placement and sub-grouping. Separation is better with IDMAP in some cases, but it tends to jam similar documents into pockets, making it more difficult to interpret neighboring relationships in dense regions.

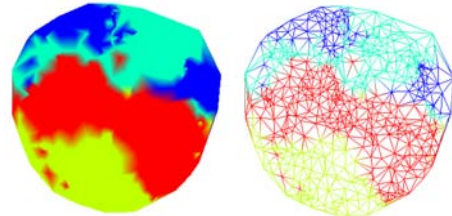
In 'more generic texts' the intrinsic content relationship is a lot less obvious than in academic or scientific papers. Figure 17 shows the generation of Kolmomaps and IDMAPs from messages in discussion groups (data sets message1 and message2 in Table 1). Pseudo-class in this case is the theme of the discussion groups. The maps show that Kolmomaps are capable of distinguishing subjects in that context much better than IDMAPs.

Once the range of themes starts to broaden and specialize, Kolmomaps tend to mix messages coming from distinct sub-groups. This can be understood in the light of the fact that discourse tends to be full of common expressions and terms, regardless of the subject under discussion. Figure 18 shows that fact.

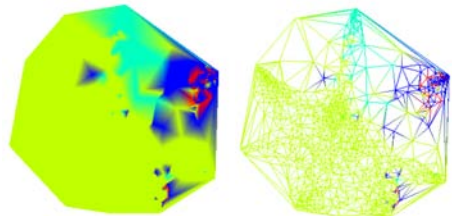
Observing the pictures in Figure 18, it can be noted that Kolmomaps can still determine various 'pockets' of similar documents and also that the mixture is not uniform, that is, at least in the body with 700 messages, there is overall mixture of 'reds' and 'blues' as well as 'greens' and 'yellows'. Blues and reds are sci.space and alt.religion, and greens and yellows are various 'comp' subjects and 'forsale' subjects. Therefore even with lesser distinction between pseudo-classes than the 3-theme case, there is an underlying pattern reflected by the Kolmogorov distances. IDMAP, as might be expected from the previous message tests, was not capable of any significant separation between subjects.



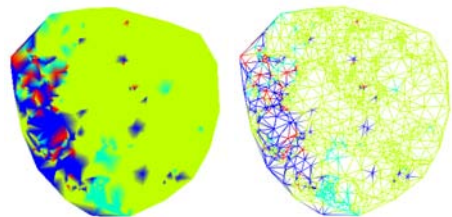
(a) IDMAP results from corpus1 set.



(b) Kolmomap results from corpus1 set.



(c) IDMAP results from corpus2 set.



(d) Kolmomap results from corpus2 set. In dataset corpus2, one part of the corpus - in yellow - 1236 files - is a lot larger than any of the other three

Figure 16: IDMAP and Kolmomap of the two bibliographic data sets. See Table 1.

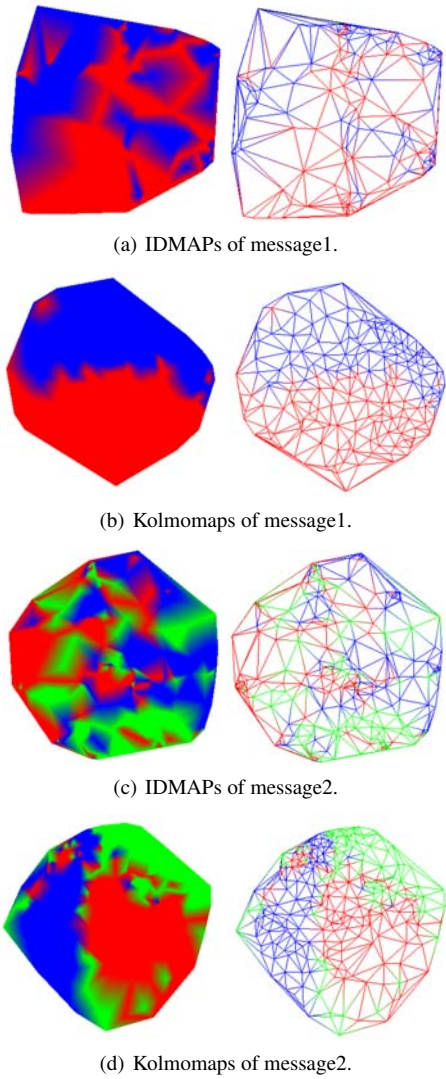


Figure 17: IDMAP and Kolmomap of the newsgroups data sets.

Detailed exploration of various maps have shown a close relationship between proximity in the map and the expected proximity of document content with few exceptions.

In order to test the capability of content based point placement, two tests were performed. In one of them 5 ‘intruders’ (documents not previously processed in the set) were added to the map. They belonged to the general class of sonification, but were not in the initial set. All five had at least two common authors and represented and evolution of the same sonification system over the years. Additionally to those, two other papers were added. Again two of the authors at least were repeated, but the main subject of the paper was not sonification. One of them mentioned sonification in-

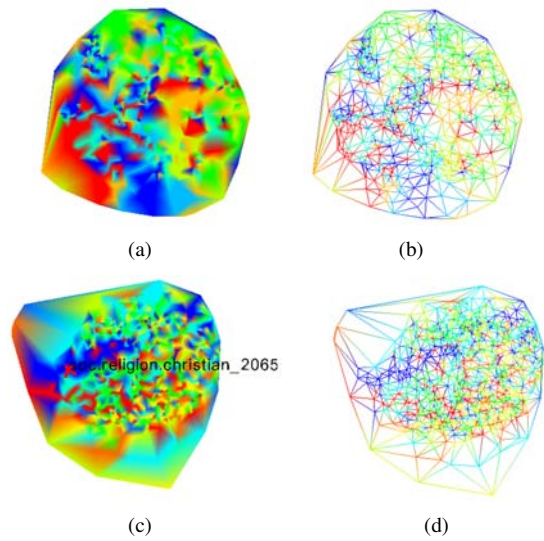


Figure 18: Kolmomaps of message3 and message4 data sets.

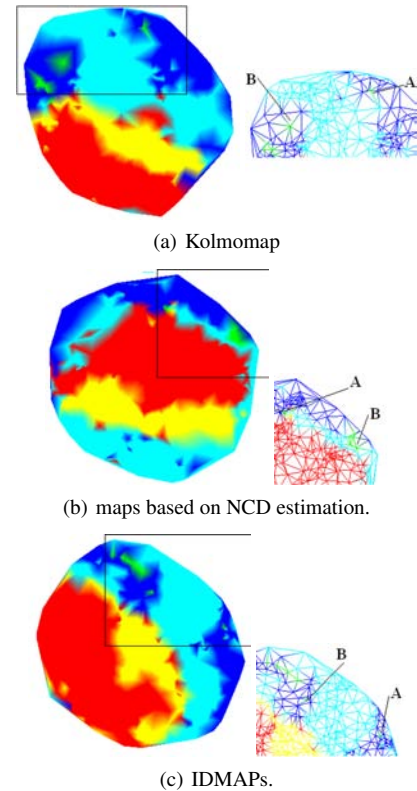


Figure 19: Maps with highly correlated ‘intruders’ (in green).

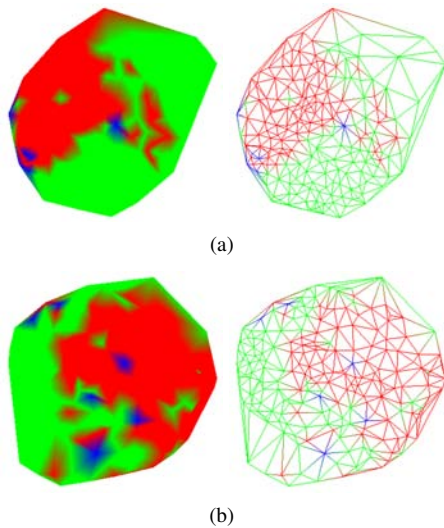


Figure 20: News-maps with uncorrelated ‘intruders’ (in blue).

centially (paper A), the other didn’t (paper B). Figure 19 shows the map for that test. In that case the Kolmogorovs projected those related papers together (Figures 19a and 19b). The papers that were not primarily on sonification were still mapped within the general sonification pseudo-class, indicating the importance of author’s names (and most likely some self-reference too) in the context. However, it can be seen that, for the k-lzw implementation, the fact that the content itself was diverse from that of the neighbors, made paper B push them away and form an isle within the group. NCD also provided approximation between most of the related papers, but provided less distinction for that text with content diverse from sonification. IDMAP did not perform as well (Figure 19a), placing the files within the scope of the general sonification groups but making less distinction of those without direct relation with sonification.

The second test meant to compare the two different approximations of Kolmogorov distances (k-lzw and NCD) further. In this test, we added intruders (also 5) in the messages test set message1 that were not related to any of the previous two newsgroups. The two groups were comp.graphics and alt.atheism and the intruders (in blue) belonged to the theme of rec.sports.baseball. Figure 20 presents the results of that placement. It shows that for the k-lzw map the new points are pushed away from the previously existing sets of messages (either to the border of the map, or, in one case, between the two groups). NCD did not perform as well, mixing half of the intruders within either one of the previously formed groups.

Projection errors according to calculations published in a previous work [TMN03] are presented in Table 2. Those values support the evidence that the mixture in maps of larger

Table 2: Approximate Projection Errors

| Set | k-lzw | NCD | IDMAP |
|----------|-------|------|-------|
| corpus 1 | 0.25 | 0.33 | 0.2 |
| corpus 2 | 0.25 | 0.32 | 0.17 |
| message1 | 0.17 | 0.25 | 0.21 |
| message2 | 0.17 | 0.25 | 0.21 |
| message3 | 0.23 | 0.3 | 0.18 |
| message4 | 0.17 | 0.25 | 0.17 |

human communication files (news groups with larger number of subjects and messages) is caused by message content instead of the projection itself.

Assigning attributes to the map components (vertices, for instance) can show one or more additional degrees of information on top of the landscape map. Various attributes can be visually mapped to color, height, isocurves, etc., allowing further analysis of relationships between the documents represented in the map. Previous maps in this text have shown pseudo-class by using color. Figure 21a shows the same data mapped redundantly to height for the corpus1 dataset. In that case, different classes are ‘plateau layers’ in the map and misplaced (or pseudo-misplaced as the case may be) points can be easily located as peaks or depressions in the topography of the maps.

To highlight proximity relationships further, one attribute mapping found useful was to perform a hierarchical clustering of the projected data. A hierarchical clustering can group points closer together in various levels. By doing that, and then showing the depth of the cluster on the maps, it is possible to have visual aid to locate pockets of closely related documents. Figure 21b illustrates the corpus1 data set after submission to single-link hierarchical clustering. Both color and height are mapping the depth of the document in the clustering tree. The dark blue (and highest) areas show where the points are closer together; the following levels or grouping are lighter blue, green, orange, yellow, and red (lowest), the latter showing the points (documents) that are most isolated from their ‘neighbors’.

The former mappings can be combined to produce composed visualizations of multiple attributes. Figure 21c shows clustering mapped to height and class mapped to color for the same dataset, and Figure 21d shows the mapping inverse to that (height is class, color is cluster).

The map can be interacted with for exploration. In our case, a tool for exploration of that type of representation, called Spider Cursor, is under development. The current version allows walking over the surface using a cursor that highlights the edges to neighbors (see Figure 21e). It also allows the user to choose how numeric attributes will be mapped to visual attributes such as color, height, level curves, glyphs and sound. Paths can be marked, selected and cut to extract parts of the map (Figure 21f). One attribute, possibly literal,

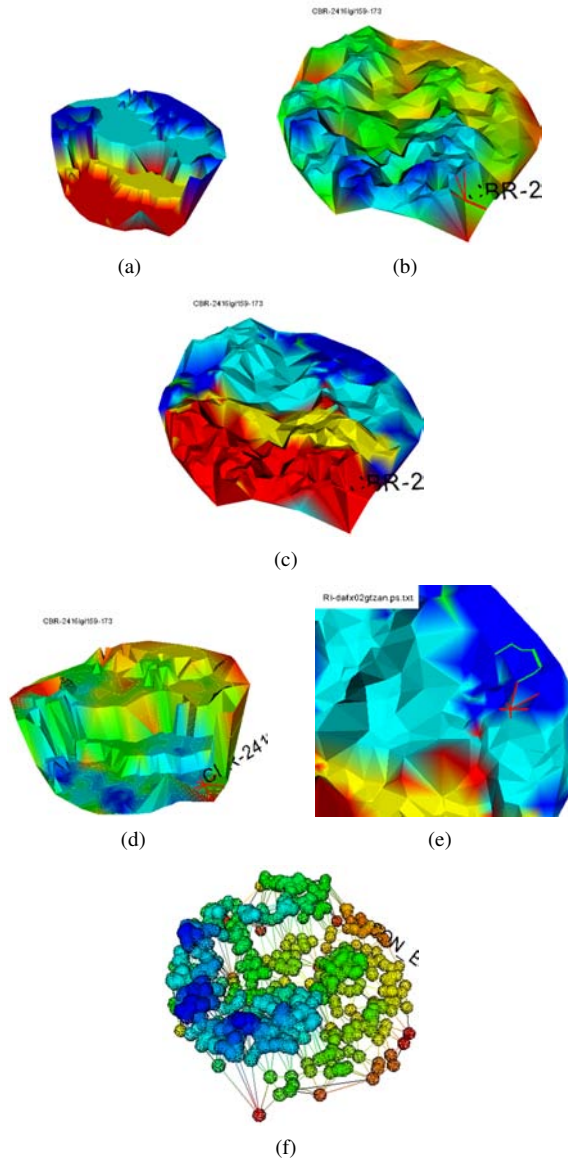


Figure 21: Various possibilities of attribute mapping and interaction with the corpus1 map.

is shown on a window on the map. In our examples, they were the labels of the files containing the document.

In visual terms, Kolmomaps were very successful both in distributing documents in larger subject areas and in grouping files with similar contents. The fact that it does not require term analysis makes it a good option to process (and organize) corpora of documents to be analyzed in an interactive session.

The process of forming the distance matrix for a whole corpus is costly, particularly if processed from scratch. Ta-

Table 3: Times for distance calculation (*h* - hours, *m* - minutes)

| Set | Processor Pentium IV - 3GHz | |
|----------|-----------------------------|------|
| | k-lzw | NCD |
| corpus 2 | 11.3 h | 5.5h |
| message3 | 2h | 17 m |
| message4 | 4.2h | 39 m |

ble 3 shows the times to process each data set in full at once, with no intermediate storage (that is, the matrix was not built incrementally). The projection itself, including hierarchical clustering, is done in a matter of few seconds.

A.6. Conclusions

The technique presented here of similarity calculation via Kolmogorov complexity taken from raw texts has proven useful in conjunction with distance-based projections to build maps of texts. Those distances have shown to separate general content areas and to generate proximity between similar texts, facts reflected by the projection based on the distances.

Kolmomaps have shown to work well for texts in a variety of contexts. It is a good step towards helping text organization by content in groups of documents in a context, where selection of pre-filtered reading material is necessary (such as research, education, training in technology etc.).

Kolmomaps have high computational cost in its distance calculation stage. However, some facts about the time to generate maps of documents based on distances by Kolmogorov complexity estimation should be phrased:

- No pre-processing is needed. The approach allows immediate application to text documents without the need to realize (or understand) vector transformation. No lengthy editions are necessary either. That in itself saves time in the whole analysis process.
- Opposed to vector representation schemes, this process is incremental. For data sets that grow – and they usually do –, it is possible to store intermediate distances and symbol tables, speeding up the calculation for a new text to be added.
- The projections are actually very fast, in the order of a few seconds (including improvement, hierarchical clustering and display).
- Although NCD is faster, k-lzw estimation has resulted in more consistent map behavior. It was also noted that k-lzw time increase showed more regular behavior than NCD as the document bodies grew larger.

The maps using k-distances compared well with IDMAPs. Interpretation of the resulting maps is easy to learn. In a short time, users learn that being close means related content, and

longer edges mean that the distance is larger than to those with smaller edges.

Further work is planned in analyzing a number of text corpora, in data structures for storing and recovering map results, and in adding extra semantic layers on top of this map to reflect other dimensions, such as co-citation and relevance. To our knowledge, the conditional Kolmogorov complexity was not evaluated our way before, so it remains to show that this measure is a (quasi-)metric, as the experimental results suggest.

Scalability to larger data sets was not an issue here. Rather, we have been researching methods to help distinguish important relationships in text data sets of a manageable size but still too large to have the user make sense of it on his/her own efficiently. However, we believe that with that the approach can be scalable a further level, by developing an incremental data organization approach to go with the mappings.

B. Technical Report: CONTENT-BASED DOCUMENT MAPS USING FAST PROJECTIONS AND TOPIC DETECTION

Fernando V. Paulovich, Rosane Minghim, Roberto Pinho, and Alneu A. Lopes

{paulovic,rminghim,rpinho,alneu}@icmc.usp.br
ICMC - University of São Paulo, Brazil

This article presents an approach to build visual maps of text collections based on their content and using multidimensional projections. A tool, called Projection Explorer (PEX) was built to support the steps of this approach. We also present two novel techniques to integrate text mining into the exploration process that help extract topics discussed within the document collection. The first is based on term co-variance and the second is based on seeded generation and weighted filtering of association rules. Topics and sub-topics in groups of documents are extracted and this is coupled with the capability of various levels of exploration of the map. This combination of techniques is demonstrated to help the user to gather a general view of the content of a document collection as well as locate and relate subtleties within a textual data set.

B.1. Multidimensional Projections for Mapping Collections of Documents

Examining text is crucial for many different types of applications. Even applications that rely on data other than textual (such as images, signals, simulations) sometimes have alternative text-based output. The troublesome task of interpreting content and extracting useful information from a document collection is the target of efforts in various areas of computer science. The combination of Text Mining and Visualization (or Visual Text Mining — VTM) is concerned with developing tools to support users extracting meaning, drawing conclusions and making decisions without extensive reading, since it is hardly possible to rely on actually going through content in detail due to time restraints found in most applications.

We describe here recent results towards building visual maps of documents based on their content through an integration of mining and visualization algorithms and techniques. The underlying approach taken here defines a placement of the documents on a plane taking into account their content.

Based on the computation of similarity amongst textual documents, they are projected as points onto a two-dimensional space with the target that similar documents should be projected close to each other and that dissimilar documents should end up far apart.

In order to create this graphical representation, a multidimensional projection technique is used. Formally, let $X = \{x_1, x_2, \dots, x_n\}$ be a set of m -dimensional data, with

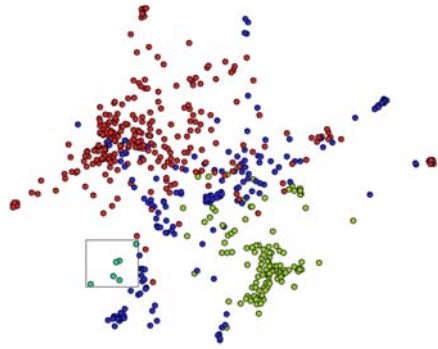
$\delta(x_i, x_j)$ a dissimilarity (distance) measure between two m -dimensional data instances, and let $Y = \{y_1, y_2, \dots, y_n\}$ be a set of points into a p -dimensional space, with $p = \{1, 2, 3\}$ and $d(y_i, y_j)$ a (Euclidean) distance between two points of the projected space. A multidimensional projection technique can be described as a bijective function $f: X \rightarrow Y$ that seeks to make $|\delta(x_i, x_j) - d(f(x_i), f(x_j))|$ as close to zero as possible, $\forall x_i, x_j \in X$ [TMN03].

The result is a planar mapping of a point set X representing the documents (we call this 2D placement a *map*), which can be used to explore the document collection represented by X . As it is going to be illustrated in the course of this text, the techniques we have developed are capable of mapping documents in such a way that text data dealing with common subjects form groups that are visually apart from other groups, thus allowing identification of patterns in the text set. An example of such a map is given in Figure 22. It shows a document map of a collection of 574 scientific papers belonging to three different subjects, previously manually classified. In the pictures, different classes have different colors (red is Case-Based Reasoning (CBR), blue is Information Retrieval (IR), and green is Inductive Logic Programming (ILP)). The small greenish group in the rectangle shows the placement of five papers on sonification (the use of sound to display information). That particular group refers to five papers reflecting the evolution of a sonification system we have developed, and are strongly correlated. Figure 22a) shows that they are placed adequately in the same neighborhood. Figure 22c) demonstrates that, when mapped in conjunction with the remaining papers, this group was placed in the outer boundary of the map and nearby papers in the data set whose main subject was audio retrieval (they are the points with colors other than red in Figure 22c). Another aspect that can be noted in the map is the fact that the groups of CBR and ILP are more compact, whilst the IR set is spread out on the map, a visual analogy to the characteristics of the subject areas, the first two being more mature and therefore having a set of own nomenclature and techniques, whereas IR relies on resources from various sources, including those of ILP and CBR.

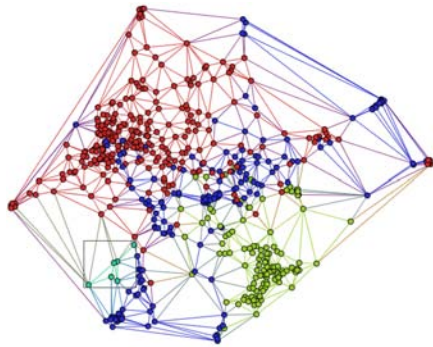
Multidimensional projection techniques can be classified into two major groups, according to the functions f employed: *linear projection techniques*; and *nonlinear projection techniques*.

Linear projection techniques, such as *Principal Component Analysis (PCA)* [Jol86], create linear combinations of the data attributes, defining them in a new orthogonal basis of small dimension. Although this type of technique performs well on Gaussian data, in handling data with nonlinear structures, such as clusters of arbitrary shapes or curved manifolds, it typically fails to capture relevant patterns. In such cases, nonlinear techniques are better candidates.

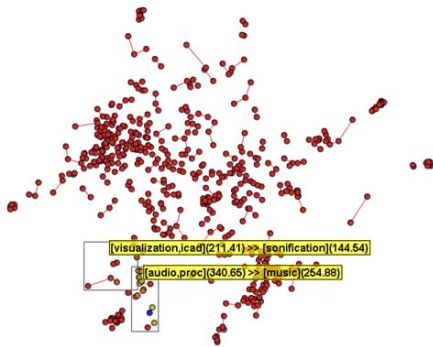
Rather than relying on linear combinations of the attributes, nonlinear techniques attempt to minimize a func-



(a) Point Placement for the text collection. Detail is the sonification test group.



(b) Delaunay Triangulation of Point placement in Figure B.1 .



(c) Placement of sonification papers midst papers of audio information retrieval papers

Figure 22: Document map of a collection of scientific papers belonging to three different main subjects, manually classified, plus a test group (inside the rectangle).

tion of the information loss incurred in the projection. Normally, this function is based on the dissimilarities amongst the m -dimensional instances and on distances among the p -dimensional points.

One example of a nonlinear projection technique is *Multi-dimensional Scaling (MDS)* [CC00]. Sprang from the psy-

chophysics domain, MDS actually comprises a class of techniques aimed at mapping instances belonging to an m -dimensional space into instances on a p -dimensional space ($p \leq m$), striving to keep some distance relations.

Amongst the various MDS techniques, the simplest ones are those based on *Force-Directed Placement (FDP)* [FR91, Cha96]. Originally proposed as a graph drawing heuristic, the FDP model aims at bringing a system composed of instances connected by imaginary springs into an equilibrium state. Instances are initially placed randomly and the spring forces iteratively push and pull them until reaching an equilibrium. To apply the FDP model as an MDS technique the spring forces must be proportional to the difference between the dissimilarity $\delta(x_i, x_j)$ among the m -dimensional instances, and the distances $d(f(x_i), f(x_j))$ among the p -dimensional points.

We have developed two distinct nonlinear projection techniques: *Projection by Clustering (ProjClus)* [PM06] and *Least-Square Projection (LSP)* [PNML06].

The core idea of ProjClus is to create and project clusters of m -dimensional points. After initial clustering, the ProjClus algorithm calculates the centroids of these clusters, and projects these centroids onto the plane. Next, it separately projects each set of points defined by the clusters to the plane; and finally the technique assembles the final layout positioning the clusters' projections according to the projection of their centroids.

The LSP technique is a generalization of an approach for mesh-recovering and mesh-editing in order to deal with high dimensional spaces. In this technique, a subset of m -dimensional points are projected onto the plane, and the remaining points are projected using an interpolation strategy that considers only the neighborhood of the m -dimensional points.

ProjClus and LSP are high-precision and fast projection techniques that are suitable to handle points belonging to nonlinear sparse spaces — such as the one generated from document vector representations (see Section B.2.3) — since they take into account neighborhoods of the original points.

Both LSP and ProjClus are available within a tool called Projection Explorer (PEX), available at: <http://www.lcad.icmc.usp.br/~paulovic/peX>. PEX makes available a set of tools to support every step and parametrization of the maps presented here, as well as a number of graph based and text based exploration tools. After high quality projection, exploration based on document content can take place in a more friendly environment. For that to occur, we have developed a number of novel techniques for helping the user to extract the main subjects being discussed and to focus on a particular area to locate details of interest.

B.2. Exploring Content-based Document Maps

This section presents two novel techniques (and its support interactive tools) for visual analysis of content based on topic extraction. Covariance and Generation of Association Rules are the basis for these techniques.

B.2.1. Topic Identification by Covariance

Although a projection can help locating groups of related documents, the initial map does not reveal information within these groups. In order to extract information about these groups, a group can be selected and a label can be generated which aims at identifying the main topics discussed within that group. A covariance-based label is composed by the three distinct terms that have the highest covariance considering only the selected documents. The covariance of two terms is a measure of the degree to which they vary together. The covariance is calculated according to Equation 6.

$$\text{cov}(t_i, t_j) = \frac{1}{n-1} \sum_{k=1}^n (t_{ki} - \bar{t}_i)(t_{kj} - \bar{t}_j) \quad (6)$$

where \bar{t}_i is the mean of the i^{th} term t_i , and t_{ki} and t_{kj} are the values of the i^{th} and j^{th} terms for the k^{th} document.

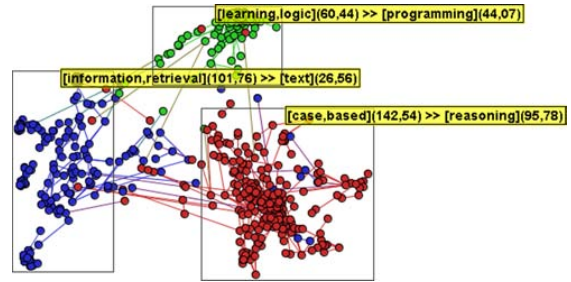
Figure 23 shows another map of the same data set of Figure 22, showing the resulting labels after group selection by the user.

The first two terms on the left side of a label are the ones that present the greatest covariance. The third one on the right side is the term which has the greatest *mean covariance* according to the labels already chosen. The mean covariance is calculated as the mean between the covariances of a term taking two other terms. The numbers between parenthesis are the covariances.

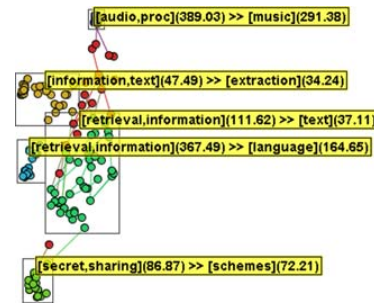
If the covariance approaches zero, it means that the terms are independent. On the label, this number can indicate if the three terms are equally related or if only the first two are. If the number on the left side is much greater than the number on the right side, it means that the third term is not deeply related to the other two. But if these numbers are close, it means that the three terms occur frequently together.

For instance, in Figure 23(b), considering the label $[\text{learning, logic}](60.44) \gg [\text{programming}](44.07)$ and dividing the right covariance by the left we get $\frac{44.07}{60.44} \approx 0.73$. Calculating the same rate for the label $[\text{information, retrieval}](101.76) \gg [\text{text}](26.56)$, we get $\frac{26.56}{101.76} \approx 0.26$. Thus, it is possible to infer that the terms “learning” and “logic” are much more related with “programming” than the terms “information” and “retrieval” are with “text”.

Notice that the covariance is not a causality relation between terms, that is, it does not indicate that one term occurs due to the other one. It is only a measure of the degree to



(a) Document map of a scientific papers' collection in three different subjects.



(b) A view of part of the map in Figure 23(a).

Figure 23: Example of labels for document maps.

which two terms vary together. For example, on the map of Figure 23(b), which is a view of part of the Figure 23(a), the terms “secret” and “sharing” present a high degree of covariance, but it is not possible to establish that the term “secret” will cause the term “sharing” or the other way round. Another important aspect is that using two terms only, it is not normally possible to direct infer why they are related. This is the point where the third term of the label can help. The third term for the terms “secret” and “sharing” is “schemes”, thus it is possible to conclude that the main topic of the document used to create such label is somehow related to cryptography.

Inside PEX there are two ways to define the groups of documents used to generate labels. Either the user selects a rectangular area on the projection or a clustering of the documents after projection is performed to suggest a particular grouping and generate labels all at once. Two types of clustering are available in PEX, k-means and density based clustering. The labels presented on Figure 23(b) were automatically created using the density based clustering approach.

Figure 24 presents a map of a data set composed by text files including titles, authors and abstracts of all papers published by IEEE InfoVis Conference from year 2000 to 2005 and by the International Conference on Information Visualisation (IV), also published by IEEE CS Press, from year 2000 to 2006. 1020 textual documents are mapped. Some selected groups and their labels by co-variance are presented

in that picture. Darker nodes are InfoVis papers and lighter ones are IV papers. It could be observed that doing a search within the map, papers about graph based visualization, image based techniques, network, web and various others subjects concentrated on small neighborhoods in the map. Also, a certain distribution of subjects between the conferences can be observed, revealing the types of paper presented in each.

The process employed to explore a document map using covariance-based labels is a step-wise refinement process. For example, the view presented in Figure 23(b) is labeled as $[information, retrieval](101,76) \gg [text](26,56)$ in Figure 23(a). Thus, when such group is explored in detail, the words “information” and “retrieval” can be ignored for calculating the covariance since we already know that these documents deal mainly with “information” and “retrieval”. The third term “text” is not ignored in this analysis since the right covariance is much lower than the left one. In the case where the two covariances present similar values, the three terms can be discarded. Lists of terms to be ignored when extracting topics can be edited by users of PEX.

Although co-variance based topic extraction is very useful for a first view of the possible subjects around the map, further information can be automatically extracted to support revealing subtleties and laying out more subjects under discussion in a particular group. The next technique contributes to achieving precisely that goal.

B.2.2. Topic Extraction by Sequential Covering Induction of Association Rules

An appropriate set of association rules derived from a collection of text documents can be used to describe a context in which a term appears or also the context or topic related to a subset of documents. When mining association rules from text, an association rule (AR) is an implication of the form $X \Rightarrow Y$, where $X \cap Y = \emptyset$ and they are both subsets of L , where $L = l_1, l_2, \dots, l_m$ is a set of literals or items (each item representing a term from the bag of words). A transaction T is a set of items $T \subseteq L$ that represents a document from the corpus C . The rule $X \Rightarrow Y$ holds in the document set C with confidence c if $c\%$ of the documents in C that contain X also contain Y . The rule $X \Rightarrow Y$ has support $s\%$ in C , if $s\%$ of documents in C contain $X \cup Y$.

A rule $R_i: X \Rightarrow Y$ with significant support in C means that a set of terms $l_i \in X \cup Y$ are found together in a large subset of documents from C . Moreover, if R_i has high confidence, then the occurrence of X (the body) means high probability for the occurrence of Y (the head), at least in C .

In a previous work [LPMP07], we presented an algorithm to produce and select good association rules in order to describe the main subject or topic in a selected set of documents S_k . That algorithm deals with the problem of the large number of association rules as follows. Instead of post-pruning the set of rules generated from all documents in S_k

by some rule quality measure, it generates rules which contain at least one term (seed) present in a selected set of most relevant terms. The term relevance is given by a weight that favors terms with higher frequency in the selection than in the rest of the corpus. An arbitrary number of rules, usually 1–3, with the highest term weight summation is then selected for display as label for the group selected. Eventually, some documents carry topics that are not described by these selected rules, that is, they are not covered by the rule.

Here, we present a new algorithm that uses a sequential covering strategy in order to extract topic targeted rules from the documents uncovered by the main rule. Additionally, instead of just selecting the n most weighted rules, we keep only those rules which provide additional covering over the previous ones. This reduces the amount of redundancy found in the selected rules, considering that if two rules cover the same subset of documents, only the one with the highest weight is kept.

The Algorithm (detailed below) has two nested loops. In the inner loop, the iterative algorithm for generating and ranking rules over a selected area S_k of the projection leads to a set of rules SR . This process is repeated (outer loop) removing covered documents during the iteration until no remaining document is left or the rule generation process outputs an empty set of rules. The function $coverage(AR_i, U)$ on Algorithm 1 denotes the number of documents from U that support the rule AR_i .

Even considering that each selected rule must cover documents not covered by any other previously selected rule, a considerable number of rules may be generated for a given selection of documents. To reduce cluttering, we apply a rule grouping strategy that relies both on literals in the rule and on overlap of covered documents. Intuitively, if rules share some of their literals and cover roughly the same set of documents, they are likely to represent the same topic, but described in a slightly different way. Thus, instead of presenting a label for each rule, only one label per group of rules is displayed. A rule AR_i joins a group of rules G_j , if (i) AR_i has at least one term $t \in T_{G_j}$, where T_{G_j} is the set of literals found in rules already in the group G_j , and (ii) there is an overlap between the set of documents covered by the rule, and the set of documents that are covered by any of the rules in the group. This document coverage overlap should be equal or higher than a given constant α . Here, we used $\alpha = 0.5$. For computing the overlap, we use:

$$overlap = \frac{|\cap D_{AR_i}, D_{G_j}|}{\min(\{|D_{AR_i}|, |D_{G_j}|\})}$$

where D_{AR_i} is the set of documents covered by the rule AR_i , and

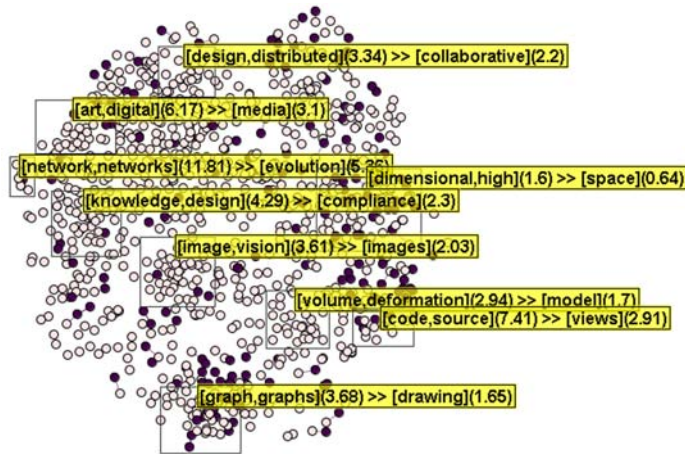


Figure 24: A map of IV and InfoVis Papers from 2000 to 2006.

Algorithm 1 - Sequential Covering of document selection

Input: S_k % k selected points
% (document \times term matrix ($M_{k,l}$))
 Corpus C % n points, (document \times term matrix ($M_{n,l}$))
 Bag_of_words % l literals of matrix document \times term
 Output: RuleSet % Set of rules that covers selection S_k

Algorithm

```

RuleSet =  $\emptyset$ 
U  $\leftarrow$   $S_k$  % uncovered documents
do
    SR  $\leftarrow$  call iterative generation and rank of rules over U [LPMP07]
    NR  $\leftarrow$  |SR| % cardinality of SR
    //rule selection
    while(SR  $\neq$   $\emptyset$ )
        remove most ranked rule  $AR_i$  from SR
        if coverage( $AR_i, U$ ) > 0
            RuleSet = RuleSet  $\cup$  { $AR_i$ }
            U = U - Set of documents covered by  $AR_i$ 
        end if
    end while
while(U  $\neq$   $\emptyset$  and NR > 0)
return RuleSet
    
```

$$D_{G_j} = \bigcup_{AR_i \in G_j} D_{AR_i}$$

If a rule does not join any group, a new group is formed with that rule as an initial element. The label for a group of rules begins with the highest weighting term taken from the rule with the highest support in the group. The label shows also this high support rule followed by all the terms that are found in the other rules of the group and that have not yet appeared.

For instance, the label “mosque:[suicide] \leftarrow [mosque], (27/20%;90%):iraq;baghdad:” is given to a rule group where the rule “[suicide] \leftarrow [mosque]” has the highest support among the rules in that group (27 documents or 20% of selected points S_k), “mosque” is its term with the highest weight, and 90% is its confidence. In the example, “iraq” and “baghdad” are the terms found in other rules from the group that does not appear in the highest support rule.

To demonstrate the potential of this technique in aiding document collection exploration, we used a corpus that is composed by 2684 RSS news feed articles, collected from

BBC, Reuters, CNN, and Associated Press sites, during two days in April 2006. Figure 25 presents the whole set of documents colored according to a clustering algorithm in PEx. Labels have been generated for each of the clusters and a few of them were selected for display.

Figure B.2.2 zooms in on a region that has almost all results for the search “iraq”. Over that display, a user selected two regions that were richer in documents with “iraq”. The labels shown (Figure B.2.2) reveal that the two regions indeed refer to the topic “iraq” from different perspectives. The region on top clearly deals with terror attacks, while the one on the bottom is more concerned about political and legal aspects of the issue. Further inspecting the top region, the user then selected two small clusters of documents inside it, which allowed him to discover that each of them dealt with a particular terror attack (Figure B.2.2). In the latter figure documents were colored using terms from the two main rules from the display labels in a boolean search query (“(mosque and suicide) or (car and bomb)”). This was done by a text search operation of PEx, in order to highlight documents covered on both selections. The same is true for individual rules, if the user chooses not to use the rule grouping strategy. Using PEx, the user can inspected one or several documents from the selected regions and confirm what has been hinted by the labels.

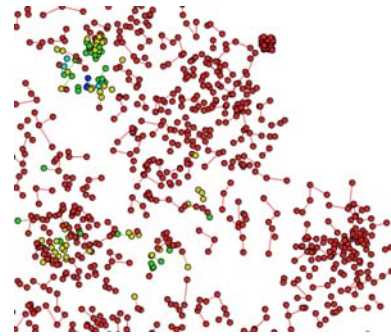
In the following text, the complete process from text to interaction supported by these tools is summarized.

B.2.3. Overview of the Mapping Generation Process

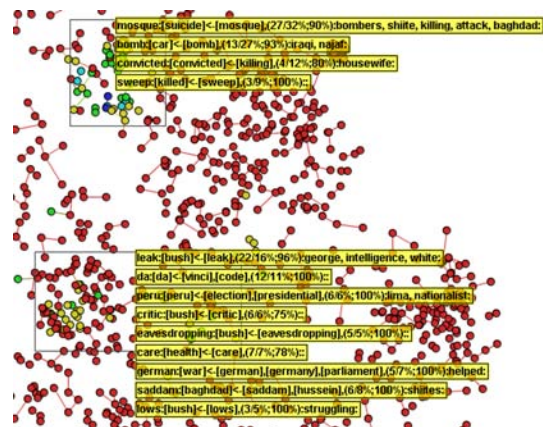
This section provides some details on the steps to build a map from a document collection by PEx. From the data set (given as separate files for each document in ASCII format), the process followed to generate and explore content-based document maps is summarized in Figure 27. This process is composed by two main tasks, *creating the map* and *exploring the map*

After text data collection, the documents are pre-processed (1) in order to establish their vector representation. This process converts each document into a vector in a multidimensional space (each term or combination of term representing a dimension), and uses a vector-based distance metric to compute the dissimilarities between documents. The steps to accomplish text pre-processing are [WIZD04]:

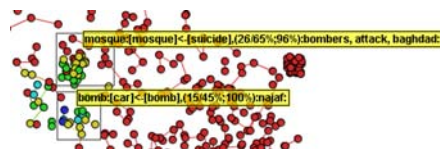
1. Non-representative words (stopwords) are eliminated from the documents, such as prepositions, articles, and other words not relevant to context;
2. The remaining words are converted into their radicals (stemming) using *Porter's* stemming algorithm;
3. The terms are counted in each document in order to determine their frequencies;
4. Luhn's upper and lower cut-off are applied to eliminate terms that are too frequent and too rare
5. The terms frequencies on each document are weighted



(a) Color by search “iraq”. Red documents have no occurrence of the word.



(b) Two user-selected regions with corresponding labels for groups of rules.



(c) Detailed Zoom and selection of two regions. Documents colored by search to highlight documents that support either the rule “mosque ← suicide” or “car ← bomb”

Figure 26: Zoom and selections over document map of 2684 RSS news feed articles.

according to *term-frequency inverse document-frequency* (*tfidf*) measure. In *tfidf*, if a term appears in almost every document, its “representation power” is decreased since such term is not capable of distinguish individual documents on the collection.

6. The vectors formed are normalized in order to have unity size. Thus, documents that have different sizes but similar content can be correctly compared.

PEX offers various visual tools in order to help the user to choose the parameters of this documents process.

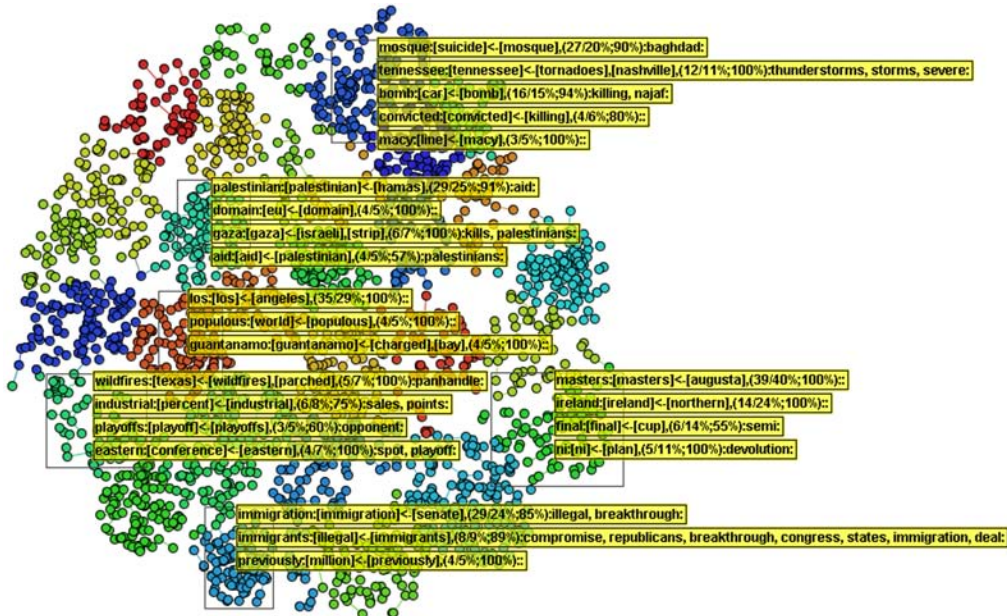


Figure 25: Document map of 2684 RSS news feed articles, collected from BBC, Reuters, CNN, and Associated Press sites, during two days in April 2006. Projection by ProjClus. Individual colors denotes clusters of documents. Shown labels were generated from groups of rules induced using some of the clusters of documents.

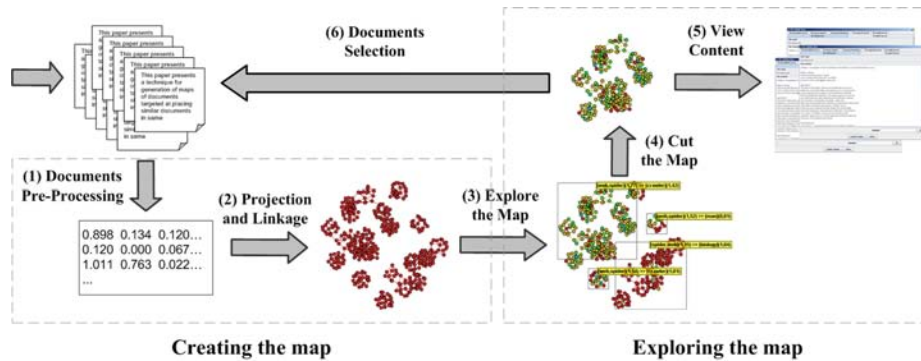


Figure 27: The suggested process which can be used with PEx to help the user to identify useful reading material.

The final result of the vector representation is a document x terms matrix coding documents' coordinates. From that matrix, a measure of distance between two documents is determined using a cosine-based metric applied on the vectors that represent them [FL95]. This kind of metric is preferred instead of other most common metrics, such as Euclidean, since this is less sensitive to the sparsity of the space, and the document x terms matrix normally is very sparse.

The multi-dimensional projection is then generated from such calculated distances.

Projected points can be connected to in various forms to help explore the map. In PEx, a triangulation or a connec-

tion by next neighbors of any order can be accomplished. Attributes can be mapped to color and size of the nodes, and node content can be loaded as illustrated above. Various graph manipulation tools are available and point placement after projection can also be performed based on any of the available edge determination alternatives.

The map can be updated by eliminating nodes and groups of nodes, moving them and coloring by certain algorithms.

For instance, the same metric that allowed the calculation of document distances can be used to color code each node on the resulting graph based on its distance to a particular document. Figure 28 presents the same map shown

in Figure 24. In this picture, the nodes with colors different from red are the most similar to the paper entitled "Semi-Automatic Image Annotation Using Frequent Keyword Mining".

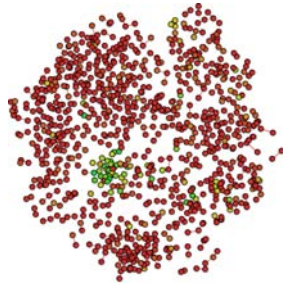


Figure 28: Color coding by distance from a particular element of the map.

The process presented here can be understood as a visual process based on content similarity amongst documents which aims at supporting the user to search, find and focus on sets of relevant documents in a large variety of applications and to find particularities of text collections.

As an alternative to the documents vector representation, an approximation of the Kolmogorov complexity has been used, called *Normal Compress Distance (NCD)* [CV05]. NCD was originally proposed as a metric between DNA sequences, but has been successfully employed to documents [TMP07]. As the main advantage, NCD does not need to pre-process the document collection in order to define the similarity between documents nor it needs information about the entire collection such as needed to calculate tfidf.

B.3. Further Remarks

The knowledge discovery process is, by its very nature, highly interactive and iterative. However, when humans are dealing with large sources of explorative information, it is very difficult even to drive the mining process. Visual text mining techniques in general and the novel ones presented here presented here are able to leverage the user interaction, speeding up the process that gives rise to useful and novel knowledge. Scanning, filtering, brushing, zooming, topic extraction and automatic labeling enable users to quickly get feedback and reformulate or adjust data preprocessing criteria in order to achieve a better display on which he/she can concentrate on.

As further steps to increase the power of the presented tools we intend to use terms drawn from the process of rule generation to build a representational space that at the same time present relevant concepts within a text data set and can offer various levels of generalization amongst texts. Various treatments of the distance measures and of the subtleties of the projections are also being studied further and formalized within a large project on visual analytics.

References


- [ABY03] ALONSO O., BAEZA-YATES R.: Alternative implementation techniques for web text visualization. In *Proc. First Latin American Web Congress (LA-WEB 2003)* (Santiago, Chile, November 2003), IEEE Computer Society, IEEE Press, pp. 202–203.
- [AF03] ASLAM J. A., FROST M.: An information-theoretic measure for document similarity. In *SIGIR '03: Proc. 26th annual international ACM SIGIR conference on Research and development in information retrieval* (New York, NY, USA, 2003), ACM Press, pp. 449–450.
- [AKS*02a] ANDREWS K., KIENREICH W., SABOL V., BECKER J., DROSCHL G., KAPPE F., GRANITZER M., AUER P., TOCHTERMANN K.: The infosky visual explorer: exploiting hierarchical structure and document similarities. *Information Visualization Journal 1* (2002), pp. 166–181.
- [AKS*02b] ANDREWS K., KIENREICH W., SABOL V., BECKER J., DROSCHL G., KAPPE F., GRANITZER M., AUER P., TOCHTERMANN K.: The infosky visual explorer: exploiting hierarchical structure and document similarities. *Information Visualization 1*, 3/4 (2002), pp. 166–181.
- [AOL93] ALLEN R. B., OBRY P., LITTMAN M.: An interface for navigating clustered document sets returned by queries. In *Proc. ACM Conf. Organizational Computing Systems* (1993), pp. 166–171.
- [BCB03] BORNER K., CHEN C., BOYACK K.: Visualizing knowledge domains. *Annual Review of Information Science & Technology 37* (2003), pp. 1–51.
- [BCG*99] BOOKER A., CONDLIFF M., GREAVES M., HOLT F., A.KAO, PIERCE D., POTEET S., WU Y.-J.: Visualizing text data sets. *Computing in Science and Eng.* 1, 4 (1999), pp. 26–35.
- [BEW95] BECKER R. A., EICK S. G., WILKS A. R.: Visualizing network data. *IEEE Tran. Visualization and Computer Graphics 1*, 1 (March 1995), pp. 16–28.
- [BY96] BAEZA-YATES M. R.: Visualizing large answers in text databases. In *Proc. Int. Workshop on Advanced User Interfaces (AVI'96)* (1996), ACM Press, pp. 101–107.
- [BYRN99] BAEZA-YATES R., RIBEIRO-NETO B.: *Modern information retrieval*. Addison-Wesley Harlow, England, 1999.
- [CC92] CHALMERS M., CHITSON P.: Bead: explorations in information visualization. In *15th Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'92)* (1992), IEEE CS Press, pp. 330–337.
- [CC00] COX T. F., COX M. A. A.: *Multidimensional Scaling*, second ed. Chapman & Hall/CRC, 2000.
- [Cha93] CHALMERS M.: Using a landscape metaphor to represent a corpus of documents. In *Spatial Information Theory: A Theoretical Basis for GIS* (1993), Frank A. U., Campari I., (Eds.), vol. 716 of *Lecture Notes in Computer Science*, Springer, pp. 377–390.
- [Cha96] CHALMERS M.: A linear iteration time layout algorithm for visualising high-dimensional data. In *Proc. 7th IEEE Visualization, (VIS)'96* (Los Alamitos, CA, USA, 1996), IEEE Computer Society Press, pp. 127–132.
- [Che73] CHERNOFF H.: *Journ. Amer. Stat. Assoc.* 68, 361 (1973).
- [Che04] CHEN C.: *Information Visualization: Beyond the Horizon*. Springer, 2004.
- [CHR03] CAREY M., HEESCH D. C., RUGER S. M.: A visualization tool for document searching and browsing. In *Proc. Intl Conf. Distri. Multimedia Sys.* (2003).
- [CLMS02] CHEN X., LI M., MCKINNON B., SEKER A.: *A theory of uncheatable program plagiarism detection and its practical implementation*. Tech. rep., UCSB, 2002.
- [CNT04] CHERFI H., NAPOLI A., TOUSSAINT Y.: Towards a Text Mining Methodology Using Frequent Itemsets and Association Rules. *Soft Computing Journal 11* (2004), pp. 431–441.
- [CPMT07] CUADROS A. M., PAULOVICH F. V., MINGHIM R., TELLES G. P.: Point placement by phylogenetic trees and its application to visual analysis of document collections (conditionally accepted). IEEE CS Press.
- [CV05] CILIBRASI R., VITÁNYI P.: Clustering by compression. *IEEE Trans. Information Theory 51*, 4 (2005), pp. 1546–1555.
- [DFK*04] DRINEAS P., FRIEZE A., KANNAN R., VEMPALA S., VINAY V.: Clustering large graphs via the singular value decomposition. In *Machine Learning* (2004), vol. 56, pp. 9–33.
- [Eic96] EICK S. G.: Aspects of network visualization. *IEEE Computer Graphics and Applications 16*, 2 (March 1996), pp. 69–72.
- [FFW91] FOWLER R., FOWLER W., WILSON B.: Integrating query, thesaurus, and documents through a common visual representation. In *14th Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'91)* (1991), pp. 142–151.
- [FL95] FALOUTSOS C., LIN K.: Fastmap: A fast algorithm for indexing, datamining and visualization of traditional and multimedia databases. In *ACM SIGMOD Intl. Conf. Management of Data* (San Jose-CA, USA, 1995), ACM Press: New York, pp. 163–174.
- [FR91] FRUCHTERMAN T. M. J., REINGOLD E. M.:

- Graph drawing by force-directed placement. *Software — Practice and Experience* 21, 11 (1991), pp. 1129–1164.
- [GSAK04] GRANITZER M., SABOL W. K. V., ANDREWS K., KLIEBER W.: Evaluating a system for interactive exploration of large, hierarchically structured document repositories. In *Information Visualization 2004* (Austing- TX, USA, 2004), IEEE CS Press, pp. 127–132.
- [H.91] H. L.: Color icons: Merging color and texture perception for integrated visualization of multiple parameters. In *Proc. IEEE Visualization'91* (October 1991).
- [HCC*97] HAN J., CHIANG J. Y., CHEE S., CHEN J., CHEN Q., CHENG S., GONG W., KAMBER M., KOPERSKI K., LIU G., LU Y., STEFANOVIC N., WINSTONE L., XIA B. B., ZAIANE O. R., ZHANG S., ZHU H.: Dbminer: a system for data mining in relational databases and data warehouses. In *CASCON'97: Proc. 1997 Conf. Centre for Advanced Studies on Collaborative research* (1997), IBM Press, p. 8.
- [Hei07] HEIJS A.: Panel paper: Requirements for coordinated multiple view visualization systems for industrial applications (to appear). In *CMV '07: Proc. 5th Int. Conf. Coordinated & Multiple Views in Exploratory Visualization* (2007), IEEE CS Press.
- [HHWN02] HAVRE S., HETZLER E., WHITNEY P., NOWELL L.: Themeriver: Visualizing thematic changes in large document collections. *IEEE Trans. Visual. and Comp. Graphics* 8, 1 (Jan-Mar 2002), pp. 9–20.
- [Hof99] HOFFMAN P. E.: *Table Visualizations: a formal model and its applications*. PhD thesis, University of Massachusetts Lowell, Computer Science Department, Lowell, MA, USA, 1999.
- [HP96] HEARST M. A., PEDERSEN J. O.: Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proc. 19th Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'96)* (1996), pp. 76–84.
- [HWR03] HUANG S., WARD M., RUNDENSTEINER E.: *Exploration of dimensionality reduction for text visualization*. Tech. Rep. Technical Report TR-03-14, Worcester Polytechnic Institute, 2003.
- [IR01] IRITANO S., RUFFOLO M.: Managing the knowledge contained in electronic documents: a clustering method for text mining. In *12th DEXA Workshop* (2001), IEEE CS Press, pp. 454–458.
- [JM04] JOURDAN F., MELANCON G.: Multiscale hybrid mds. In *IV '04: Proc. Eighth Int. Conf. Information Visualisation* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 388–393.
- [Jol86] JOLLIFFE I. T.: *Principal Component Analysis*. Springer-Verlag, 1986.
- [KGM*01] K. ANDREWS, GÜTL C., MOSER J., SABOL V., LACKNER W.: Search result visualisation with xFIND. In *Proc. Int. Workshop on User Interfaces to Data Intensive Systems (UIDIS 2001)* (Zurich, Switzerland, May 2001), IEEE CS Press, pp. 50–58.
- [KHLK98] KASKI S., HONKELA T., LAGUS K., KOHONEN T.: Websom - self-organizing maps of document collections. *Neurocomputing* 1, 1-3 (1998), pp. 110–117.
- [Koh97] KOHONEN T.: Exploration of very large databases by self-organizing maps. In *Proc. IEEE Int. Conf. Neural Networks* (1997), pp. 1–6.
- [LA00] LEUSKI A., ALLAN J.: Lighthouse: Showing the way to relevant information. In *InfoVIS* (2000), IEEE Computer Society Press, pp. 125–130.
- [LBC*01] LI M., BADGER J. H., CHEN X., KWONG S., KEARNEY P., ZHANG H.: An information-based sequence distance and its application to whole mitochondrial genome phylogeny. *Bioinformatics* 17, 2 (2001), pp. 149–154.
- [LC96] LEOSKI A. V., CROFT W. B.: *An evaluation of techniques for clustering search results*. Tech. Rep. IR-76, Department of Computer Science, University of Massachusetts, Amherst, 1996.
- [Lin91] LIN X.: A self-organizing semantic map for information retrieval. In *14th Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'91)* (1991), pp. 262–269.
- [LMMP06] LOPES A., MINGHIM R., MELO V., PAULOVICH F.: Mapping texts through dimensionality reduction and visualization techniques for interactive exploration of document collections. In *Visualization and Data Analysis 2006 — Proc. SPIE-IS&T Electronic Imaging* (San Jose, California, 2006), Erbacher R. F., Roberts J. C., Gröhn M. T., Borner K., (Eds.), vol. 6060, SPIE, p. 60600T.
- [LPMP07] LOPES A., PINHO R., MINGHIM R., PAULOVICH F.: Visual text mining using association rules. *Computers & Graphics Journal, Special Issue on Visual Analytics* 31, 3 (June 2007), pp. 316–326.
- [LV97] LI M., VITÁNYI P.: *An Introduction to Kolmogorov Complexity and Its Applications*, 2nd ed. Springer Verlag, 1997.
- [MB94] MACEDONIA M. R., BRUTZMAN D. P.: Mbone provides audio and video across the internet. *IEEE Computer* 27, 4 (April 1994), pp. 30–36.
- [MC04] MORRISON A., CHALMERS M.: A pivot-based routine for improved parent-finding in hybrid mds. *Information Visualization* 3, 2 (2004), pp. 109–122.
- [MDE87] MCCORMICK B. H., DEFANTI T. A., BROWN M. D. (EDS.): Visualization in scientific computing. *Computer Graphics* 21, 6 (1987).
- [Mer97] MERKL D.: Exploration of text collections with hierarchical feature map. In *Proc. 20th Int. ACM SIGIR*

- Conference on Research and Development in Information Retrieval (SIGIR'97)* (1997), pp. 186–195.
- [MGTS90] MIHALISIN T., GAWLINSKI E., TIMLIN J., SCHWEGLER J.: Visualizing scalar field on a dimensional lattice. In *Proc. IEEE Visualization '90, IEEE CS Press* (1990), pp. 255–262.
- [MLN*05] MINGHIM R., LEVKOWITZ H., NONATO L. G., WATANABE L., SALVADOR V., LOPES H., PESCO S., TAVARES G.: Spider cursor: A simple versatile interaction tool for data visualization and exploration. In *Proc. GRAPHITE 2005 3rd Int. Conf., Comp. Graph. and Interac. Tech. in Australasia and SE Asia* (Dunedin, New Zealand, 2005), ACM Press, pp. 307–314.
- [MPL06] MINGHIM R., PAULOVICH F., LOPES A.: Content-based text mapping using multi-dimensional projections for exploration of document collections. In *Visualization and Data Analysis 2006, Proc. SPIE-IS&T Electronic Imaging* (San Jose, California, 2006), Erbacher R. F., Roberts J. C., Gröhn M. T., Borner K., (Eds.), vol. 6060, SPIE, p. 60600S.
- [MRC02] MORRISON A., ROSS G., CHALMERS M.: A hybrid layout algorithm for sub-quadratic multidimensional scaling. In *INFOVIS '02: Proc. IEEE Symposium on Information Visualization* (Washington, DC, USA, 2002), IEEE Computer Society, p. 152.
- [MRC03] MORRISON A., ROSS G., CHALMERS M.: Fast multidimensional scaling through sampling, springs and interpolation. *Information Visualization* 2, 1 (2003), pp. 68–77.
- [MTS91] MIHALISIN T., TIMLIN J., SCHWEGLER J.: Visualizing multivariate functions, data and distributions. *IEEE Computer Graphics & Applications* 11, 3 (1991), pp. 28–34.
- [MWBF98] MILLER N. E., WONG P. C., BREWSTER M., FOOTE H.: Topic islands — a wavelet-based text visualization system. In *Proc. Visualization '98* (Research Triangle Park, North Carolina, United States, 1998), IEEE CS Press, pp. 189–196.
- [OL03] OLIVEIRA M., LEVKOWITZ H.: From visual data exploration to visual data mining: a survey. *IEEE Trans. on Vis. Comp. Graph.* 9, 3 (2003), pp. 378–394.
- [Paga] PAGE C. H.: <http://complearn.sourceforge.net/>.
- [Pagb] PAGE G. H.: www.gzip.org.
- [PG88] PICKETT R. M., GRINSTEIN G. G.: Iconographic displays for visualizing multidimensional data. In *Proc. IEEE Conf. Systems, Man and Cybernetics, Vol. 1* (1988), pp. pages 514–519.
- [PLOM07] PINHO R., LOPES A. A., OLIVEIRA M. C. F., MINGHIM R.: Topic extraction from document collections using visualization and locally weighted association rules. In *submitted to PKDD* (2007).
- [PM06] PAULOVICH F. V., MINGHIM R.: Text map explorer: a tool to create and explore document maps, 2006.
- [PNML06] PAULOVICH F. V., NONATO L. G., MINGHIM R., LEVKOWITZ H.: Visual mapping of text collections through a fast high precision projection technique. In *Proc. 10th Int. Conf. on Information Visualisation, London, UK* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 282–290.
- [PNML07] PAULOVICH F. V., NONATO L. G., MINGHIM R., LEVKOWITZ H.: Least square projection: a fast high precision multidimensional projection technique and its application to document mapping (provisionally accepted). *IEEE Trans. Vis. Comput. Graph.* (2007).
- [PRTV98] PAPADIMITRIOU C. H., RAGHAVAN P., TARNAKI H., VEMPALA S.: Latent semantic indexing: A probabilistic analysis. In *Proc. 17th ACM Symposium on the Principles of Database Systems* (1998), pp. 159–168.
- [PSS81] PAUL W., SEIFERAS J., SIMON J.: An information-theoretic approach to time bounds for on-line computation. *J. Comput. Syst. Sci.* 23, 2 (1981), pp. 108–126.
- [RC03] ROSS G., CHALMERS M.: A visual workspace for constructing hybrid multidimensional scaling algorithms and coordinating multiple views. *Information Visualization* 2, 4 (2003), pp. 247–257.
- [RES98] ROHRER R. M., EBERT D. S., SIBERT J. L.: The shape of shakespeare: Visualizing text using implicit surfaces. In *IEEE InfoVis'98* (1998), IEEE Press, pp. 121–129.
- [RES99] ROHRER R. M., EBERT D. S., SIBERT J. L.: A shape-based visual interface for text retrieval. *IEEE Computer Graphics and Applications* (September/October 1999).
- [RK04] RASMUSSEN M., KARYPIS G.: *gCLUTO — An Interactive Clustering, Visualization, and Analysis System*. Tech. Rep. CSE/UMN TR 04-021, Univ. of Minnesota, Dep. of Computer Science and Engineering, 2004.
- [SA98] SWAN R., ALLAN J.: Aspect windows, 3-D visualizations, and indirect comparisons of information retrieval system. In *Proc. 21st Int. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR'98)* (1998).
- [Sal91] SALTON G.: Developments in automatic text retrieval. *Science* 253 (1991), pp. 974–980.
- [Sam64] SAMMON J. W.: A nonlinear mapping for data structure analysis. In *IEEE Transactions on Computers* (May 1964), vol. C-18, pp. 401–409.
- [SB88] SALTON G., BUCKLEY C.: Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.* 24, 5 (1988), pp. 513–523.

- [SCL*99] SEBRECHTS M., CUGINI J., LASKOWSKI S., VASILAKIS J., MILLER M.: Visualization of search results: A comparative evaluation of text, 2d, and 3d interfaces. In *22nd ACM-SIGIR Conf. Research and Development in Information Retrieval* (1999), ACM Press, pp. 3–10.
- [Sku02] SKUPIN A.: A cartographic approach to visualizing conference aborner2003abstracts. *IEEE Computer Graphics and Applications Volume 22, Issue 1* (2002), pp. 50–58.
- [SN87] SAITOU N., NEI M.: The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.* 4, 4 (1987), pp. 406–425.
- [TC89] THOMPSON R. H., CROFT W. B.: Support for browsing in an intelligent text retrieval system. *Int. Journal of Man-Machine Studies* 30, 6 (1989), pp. 639–668.
- [TMN03] TEJADA E., MINGHIM R., NONATO L.: On improved projection techniques to support visual exploration of multidimensional data sets. *Information Visualization Journal* 2, 4 (2003), pp. 218–231.
- [TMP07] TELLES G., MINGHIM R., PAULOVIČ F.: Normalized compression distances for visual analysis of document collections. *Computer & Graphics, Special Issue on Visual Analytics* 31, 3 (June 2007), pp. 327–337.
- [vWvL93] VAN WIJK J. J., VAN LIERE R.: Hyperslice: visualization of scalar functions of many variables. In *VIS '93: Proc. 4th Conf. on Visualization '93* (1993), pp. 119–125.
- [Wei01] WEIPPL E.: Visualizing content based relations in texts. In *Proc. 2nd Australian Conf. User interface* (Queensland, Australia, 2001), IEEE Computer Society, IEEE CS Press, pp. 34–41.
- [Wel84] WELCH T.: A technique for high-performance data compression. *IEEE Computer* 17, 6 (1984), pp. 8–19.
- [Wis99] WISE J. A.: The ecological approach to text visualization. *J. of the American Soc. for Inf. Sci.* 50, 13 (November 1999), pp. 1224–1233.
- [WIZD04] WEISS S., INDURKHYA N., ZHANG T., DAMERAU F.: *Text Mining: Predictive Methods for Analyzing Unstructured Information*. Springer, 2004.
- [WTP*95] WISE J. A., THOMAS J. J., PENNOCK K., LANTRIP D., POTTIER M., SCHUR A., CROW V.: Visualizing the non-visual: spatial analysis and interaction with information for text documents. In *Readings in information visualization: using vision to think* (San Francisco, CA, USA, 1995), Morgan Kaufmann Publishers Inc., pp. 442–450.
- [Zam98] ZAMIR O.: *Visualization of search results in document retrieval systems*. General examination report, University of Washington, 1998.
- [ZE98] ZAMIR O., ETZIONI O.: Web document clustering: a feasibility demonstration. In *Proc. 21st Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'98)* (1998), pp. 46–54.
- [ZE00] ZAMIR O., ETZIONI O.: *Grouper: A Dynamic Clustering Interface to Web Search Results*. Tech. rep., University of Washington, 2000.
- [ZK05] ZHAO Y., KARYPIS G.: Topic-driven clustering for document datasets. In *SIAM 2005 Data Mining Conf.* (2005).


Tutorial - Visual Mining of Text Collections



Visual Mining of Text Collections

Rosane Minghim
 ICMC – University of São Paulo
 São Carlos
 Brazil


Haim Levkowitz
 IVPR – University of Massachusetts Lowell
 USA



© Copyright 2007 Rosane Minghim and Haim Levkowitz


1

Tutorial - Visual Mining of Text Collections



Visual Mining of Text Collections


- Overview, Motivation, Goals
- Test Cases
- Basic Concepts
- From Visualization to Visual Text Mining
- Visual Text Mining Using Projections and Point Placement Strategies
- Conclusions



© Copyright 2007 Rosane Minghim and Haim Levkowitz


2

Tutorial - Visual Mining of Text Collections



Overview, Motivation, Goals


- Collections of Text
 - Metadata
 - Content
- Multi-disciplinar
- Visual Data Mining
- Visual Text Mining
- Visual Analytics



© Copyright 2007 Rosane Minghim and Haim Levkowitz


3

Tutorial - Visual Mining of Text Collections



Overview, Motivation, Goals

- Applications
 - Information gathering
 - Searching
 - Detection
 - Discovering the Unexpected
 - Relating to predetermined documents




© Copyright 2007 Rosane Minghim and Haim Levkowitz

4

Tutorial : Visual Mining of Text Collections

- Applications
 - Survey
 - Study and Education
 - Patent Search
 - News finding
 - Forensics
 - ...

© Copyright 2007 Rosane Minghim and Haim Levkowitz




5

Tutorial : Visual Mining of Text Collections

- Maps of text Collections
 - Based on Relationships (Borner & Chen)
 - Co-authorship, co-citation
 - Based on Content
 - Similarity and Grouping
 - Common underlying subject
 - → Topics


© Copyright 2007 Rosane Minghim and Haim Levkowitz



6


Tutorial : Visual Mining of Text Collections

Relationships : Topic Busts and co-word



(Mane and Borner)
2004

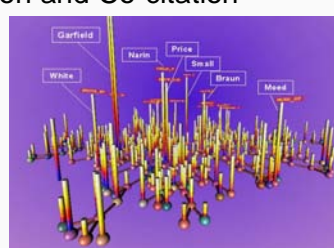
© Copyright 2007 Rosane Minghim and Haim Levkowitz



7


Tutorial : Visual Mining of Text Collections

Relationships : Citation and Co-citation



(Borner)
(2003)

© Copyright 2007 Rosane Minghim and Haim Levkowitz




8

Tutorial - Visual Mining of Text Collections

Content-based Text Mapping

- Approach 1: Dimension reduction
ex. MSD, SVD, PCA
- Approach 2: Point Placement (PP)
- Approach 3: Clustering
- Approach 4: Projections
ex. FASPMAP, NNP, LSP

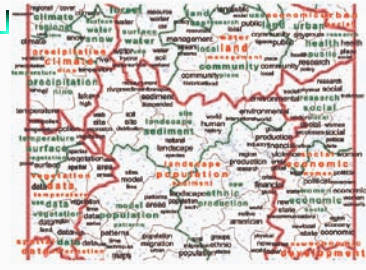
© Copyright 2007 Rosane Minghim and Haim Levkowitz



9


Tutorial - Visual Mining of Text Collections

Content - based



(Skupin)
(2002)
(abstracts)
SOM

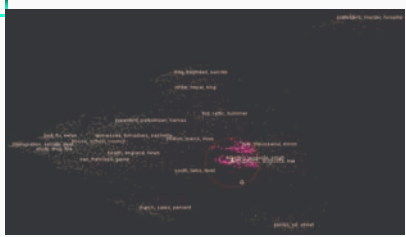
© Copyright 2007 Rosane Minghim and Haim Levkowitz



10


Tutorial - Visual Mining of Text Collections

Content - based



(Dimensional Reduction)
News flash
IN-SPIRE
(PNL)

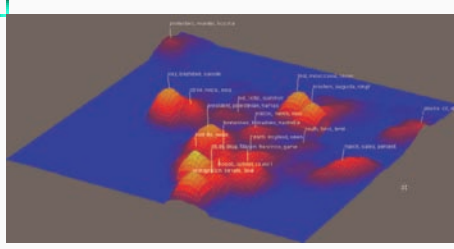
© Copyright 2007 Rosane Minghim and Haim Levkowitz



11


Tutorial - Visual Mining of Text Collections

Content - based

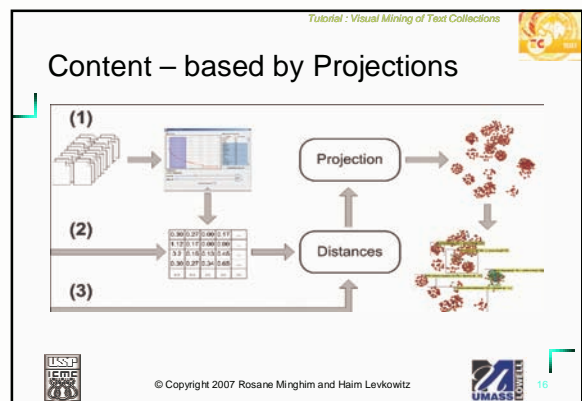
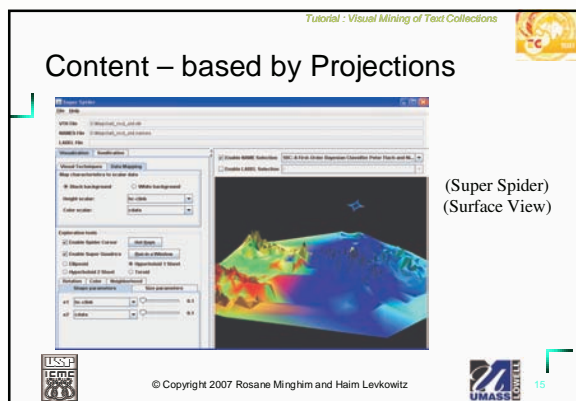
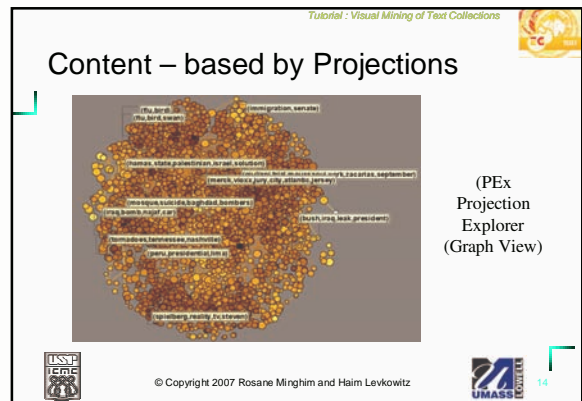
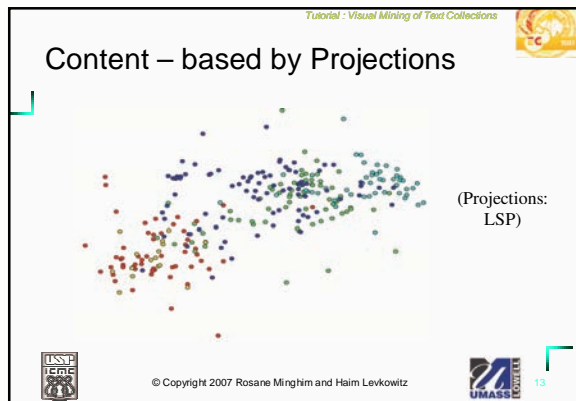


(Surface View)
IN-SPIRE

© Copyright 2007 Rosane Minghim and Haim Levkowitz



12





Tutorial - Visual Mining of Text Collections

2 – Test Cases

See main Text

© Copyright 2007 Rosane Minghim and Haim Levkowitz



17

Tutorial - Visual Mining of Text Collections

3 - Basic Concepts

- 3.1 Text Preprocessing and IR
- 3.2 Data and text mining
- 3.3 Projection techniques
- 3.4 Visual representations: graphs, surfaces, volumes, triangulations

© Copyright 2007 Rosane Minghim and Haim Levkowitz



18

Tutorial - Visual Mining of Text Collections

3.1 Text Preprocessing

1. Stopwords elimination
2. Extraction of words radicals (stemming)
3. Creation of n-grams
4. Frequency count and Luhn's lower cut (n-grams appearing less than x times are ignored)
5. Weighting process (*term-frequency inverse document-frequency - (tfidf)*)

© Copyright 2007 Rosane Minghim and Haim Levkowitz



19

Tutorial - Visual Mining of Text Collections

Result is a Vector Model

- Attributes: terms (n-grams)
- Value: term weight
- Table Data

© Copyright 2007 Rosane Minghim and Haim Levkowitz

20

Tutorial: Visual Mining of Text Collections

Vector Representation – term weighting

- tf – term frequency
- tfidf – tf x idf = tf x inverse document frequency

$$w_{ik} = tf_{ik} \times \log \left(\frac{N}{n_k} \right)$$

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Vector Representation

| | term ₁ | term ₂ | term ₃ | term ₄ | ... | term _n |
|------------------|-------------------|-------------------|-------------------|-------------------|-----|-------------------|
| Doc ₁ | 0.92 | 0.62 | 0.92 | 0.10 | ... | 0.67 |
| Doc ₂ | 0.13 | 0.11 | 1.00 | 0.34 | ... | 0.33 |
| Doc ₃ | 0.52 | 0.00 | 0.00 | 0.44 | ... | 0.77 |
| ... | ... | ... | ... | ... | ... | ... |
| Doc _n | 0.02 | 0.12 | 0.22 | 0.92 | ... | 0.00 |

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Vector Representation – Similarity calculation

EUCLIDEAN

$$sim_{i,j} = \sqrt{(w_{i,1} - w_{j,1})^2 + \dots + (w_{i,k} - w_{j,k})^2}$$

MANHATAN

$$sim_{i,j} = |w_{i,1} - w_{j,1}| + \dots + |w_{i,k} - w_{j,k}|$$

COSINE

$$sim_{i,j} = \frac{(w_{i,1} \times w_{j,1}) + \dots + (w_{i,k} \times w_{j,k})}{(\sqrt{w_{i,1}^2 + \dots + w_{i,k}^2}) \times (\sqrt{w_{j,1}^2 + \dots + w_{j,k}^2})}$$

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Vector Representation – distance calculation

$$dis(doc_i, doc_j) = \sqrt{2 * (1 - sim(doc_i, doc_j))}$$

$$sim(doc_i, doc_j) = \frac{doc_i \times doc_j}{\|doc_i\| * \|doc_j\|}$$

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Visualization (e.g.)

- Attribute Reduction
 - Co-clustering
 - PCA
 - SVD

↓ followed by

- Projection by Dimension Reduction

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Alternatively

- Similarity Calculation
 - Ex: NCD Normalized Compression Distance
 - Approximation of Kolmogorov Complexity
- Point Placement (e.g. Force-directed)

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

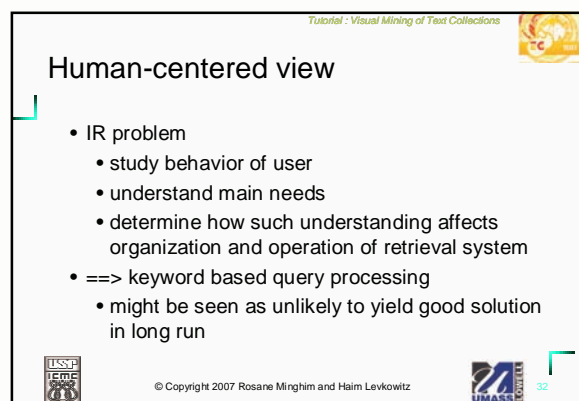
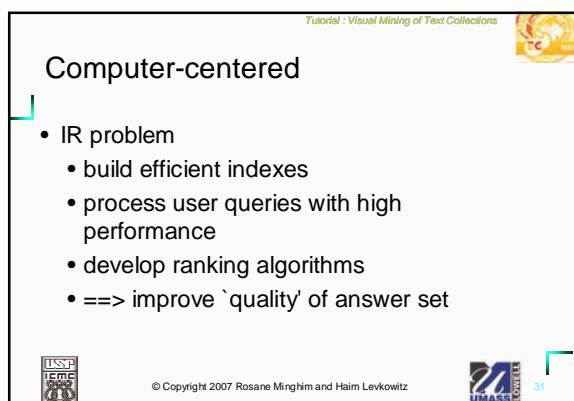
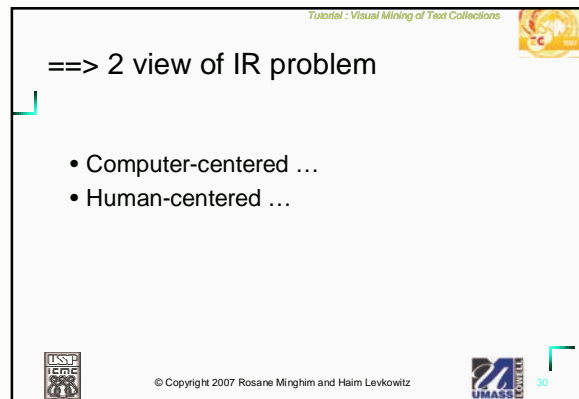
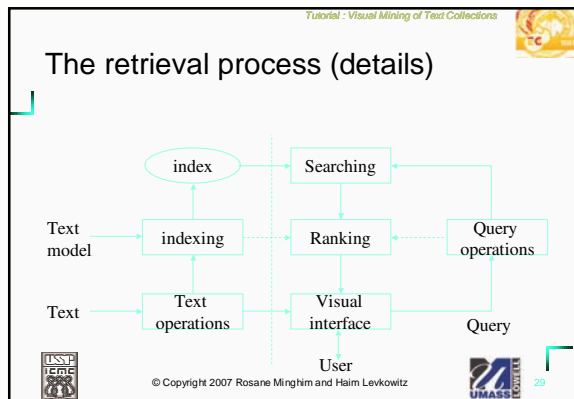
The retrieval process (high level)

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Logical view of doc

© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial - Visual Mining of Text Collections

first generation

- Basically
 - automation of previous technologies
 - E.g., card catalogs)
 - allowed searches based on
 - author name
 - title

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

33

Tutorial - Visual Mining of Text Collections

Second generation

- Increased search functionality added
- ==> search by
 - subject headings
 - keywords
 - some more complex query facilities

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

34

Tutorial - Visual Mining of Text Collections

Third generation

- Currently being deployed
- Focus on
 - improved graphical interfaces
 - electronic forms
 - hypertext features
 - open system architectures

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

35

Tutorial - Visual Mining of Text Collections

Three dramatic and fundamental changes

- First
 - access to various info sources a lot cheaper
 - ==> reaching wider audience than ever possible before

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

36

Tutorial - Visual Mining of Text Collections

Second

- Advances in digital communication
 - ==> greater access to networks
 - ==> access
 - Distantly
 - Quickly
 - few seconds

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

37

Tutorial - Visual Mining of Text Collections

Third

- Freedom to post whatever
- ==> popularity of Web
- For the first time in history
 - Most people have free access to large publishing medium

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

38

Tutorial - Visual Mining of Text Collections

- Web (and modern digital libraries) as highly interactive medium
- ==> exchange messages, photos, documents, software, videos
- `Chat` :convenient + low cost
- Can do it at time of preference
 - ==> more convenience
- ==> high interactivity
 - fundamental and current shift in communication paradigm.
 - (Searching the Web + digital libraries)

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

39

Tutorial - Visual Mining of Text Collections

Future: 3 main questions to be addressed

- First,
 - despite high interactivity, people still find it difficult (impossible?) to retrieve information relevant to their information needs
 - ==> which techniques ==> retrieval of higher quality?

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

40

Tutorial: Visual Mining of Text Collections

Second

- Ever increasing demand for access
- ==> quick response more and more a pressing factor
- ==> which techniques ==>
 - faster indexes
 - smaller query response times?

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Third

- Quality of retrieval task greatly affected by user interaction with system
- ==> how better understanding of user behavior affect design and deployment of new information retrieval strategies?

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

3.2 Data and Text Mining

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

The Knowledge Discovery Process

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

- Data Exploration is the process of **searching and analyzing** databases to **discover implicit** but potentially **useful** information

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Goals of Data Exploration

- Convey information
- Discover new knowledge
- Identify structure, patterns, anomalies, trends, relationships

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

A Confluence of Multiple Disciplines

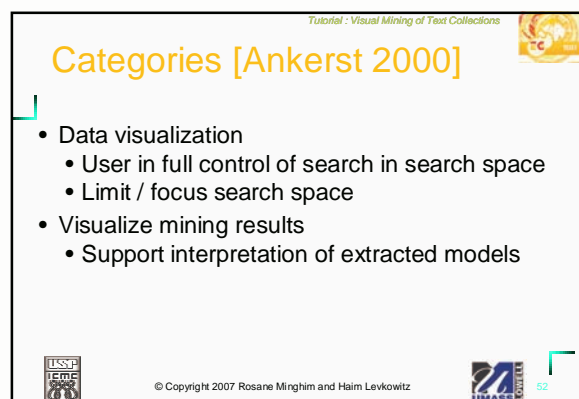
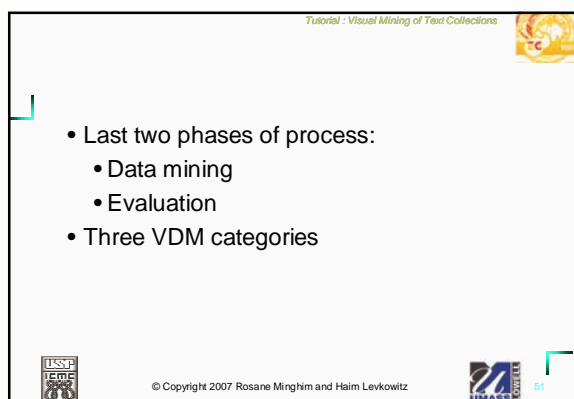
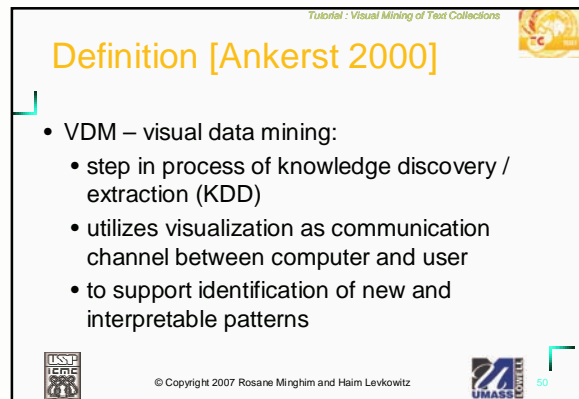
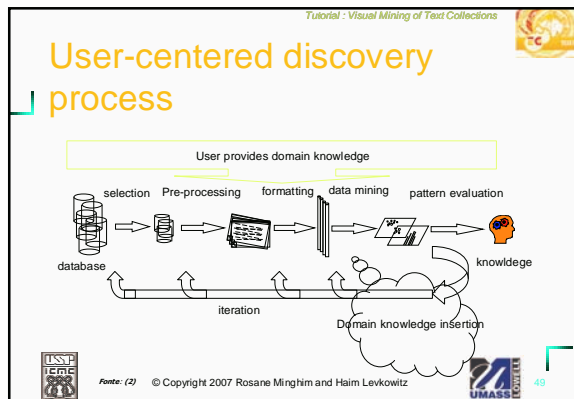
© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Data Mining Tasks & Techniques

| | | |
|--|-------------------|---|
| <p>Major Data Mining Tasks</p> <ul style="list-style-type: none"> Summarization Association Classification Prediction Clustering Time-Series Analysis | <p>using →</p> | <p>Major Techniques</p> <ul style="list-style-type: none"> Linear Regression Trees Non-Linear Regression MARS Naïve Bayes K-Means and K-Median Neural Networks |
| <p>Statistical Tools</p> <ul style="list-style-type: none"> Missing Value Imputation Normalizations Error & Variational Analysis Confidence Estimates | <p>← based on</p> | <ul style="list-style-type: none"> Association Rules Decision Trees Principal Curve Analysis Support Vector Machines Genetic Algorithms |

© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial: Visual Mining of Text Collections

- Visualize mining intermediate results
 - Guide search
 - Provide domain knowledge
 - E.g., adapt generic kernel (for different application) with user's intervention

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Categories [Ankerst 2000]

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Classification tree visualization [MineSet] -- pre

Fonte: (2) © Copyright 2007 Rosane Minghim and Haim Levkowitz

Classification tree visualization [MineSet] -- integrated

Fonte: (2) © Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Classification tree visualization [MineSet] -- pre



USSP
UMASS LOWELL

57

Tutorial : Visual Mining of Text Collections

Categories [Wong 1999]

- Loosly coupled
 - Visualization detached from analytic mining strategies
 - Support pro-processing, interpret results ...
 - Limited approach: limits of both...

USSP
UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

58

Tutorial : Visual Mining of Text Collections

- Tightly coupled
 - Visualization integrated with mining analytic strategy
 - More user control and understanding of analytic process
 - Support decision making
 - Create visual representations of search space

USSP
UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

59

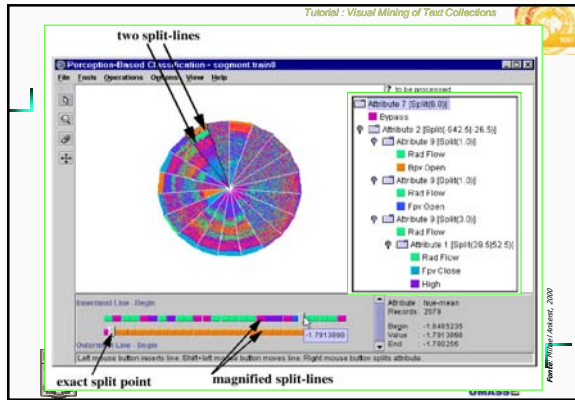
Tutorial : Visual Mining of Text Collections

- Mining with visualization
 - E.g., visual environment to generate classification trees (*Ankerst et al. 1999, 2000*)
 - Visualization with mining
 - E.g., Hierarchical Parallel Coordinates (*Fua et al. 1999*)
 - Examples with tight coupling ...

USSP
UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

60

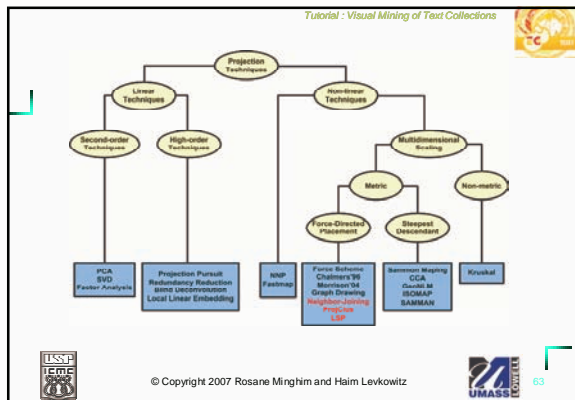


3.3 Projection Techniques

$X \in R^m \quad f \quad Y \in R^{p=\{1,2,3\}}$

- $\delta: x_i, x_j \rightarrow R, x_i, x_j \in X$
- $d: y_i, y_j \rightarrow R, y_i, y_j \in Y$
- $f: X \rightarrow Y, |\delta(x_i, x_j) - d(f(x_i), f(x_j))| \approx 0, \forall x_i, x_j \in X$

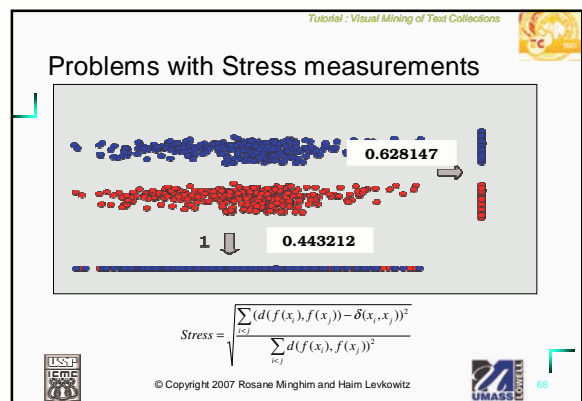
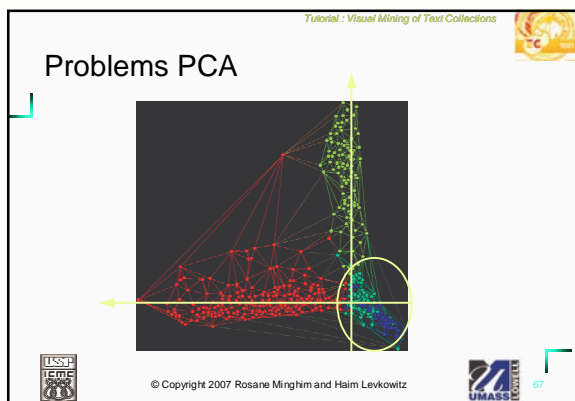
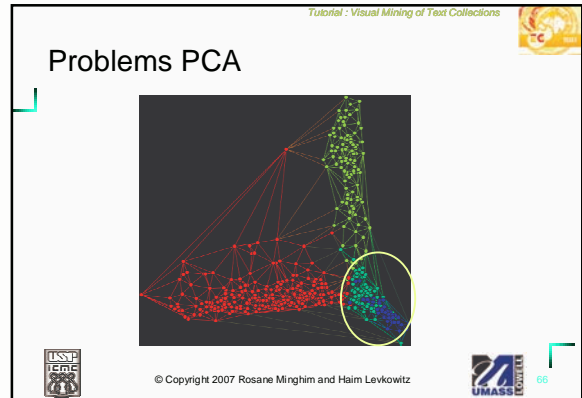
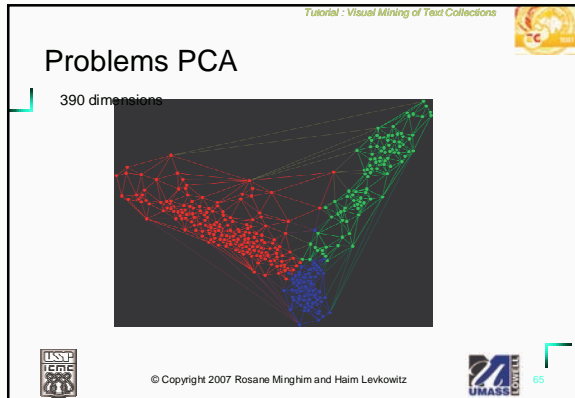
© Copyright 2007 Rosane Minghim and Haim Levkowitz

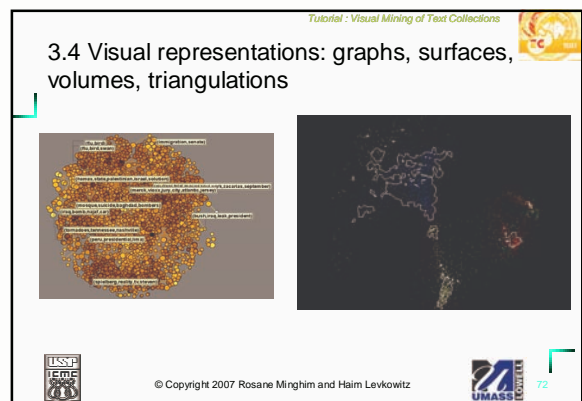
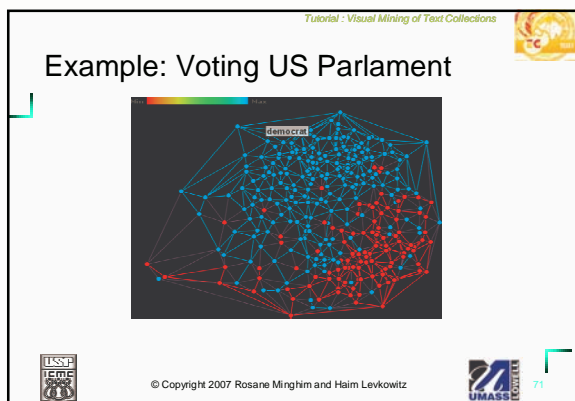
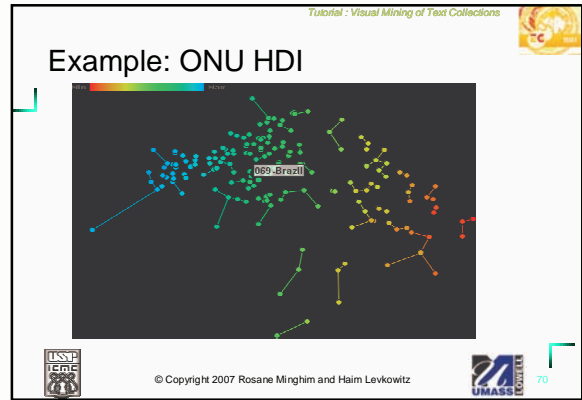
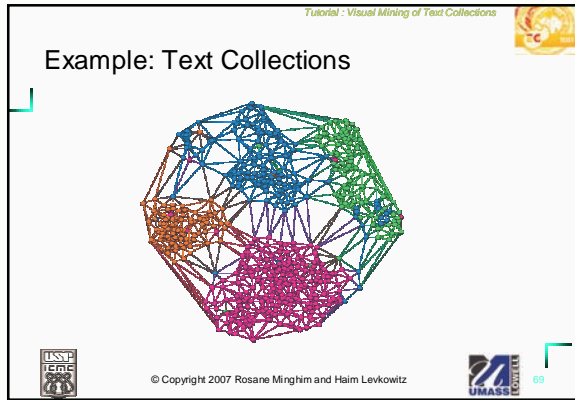


Point Placement Strategies

The diagram shows a central point x' with several vectors originating from it. The vectors are labeled with $\Delta < 0$ and $\Delta > 0$, indicating the sign of the change in distance from the point x' to other points in the space.

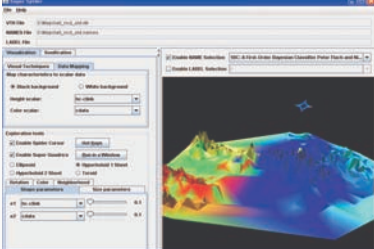
© Copyright 2007 Rosane Minghim and Haim Levkowitz





Tutorial: Visual Mining of Text Collections

3.4 Visual representations: graphs, surfaces, volumes, triangulations

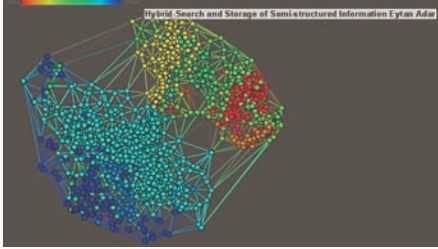


© Copyright 2007 Rosane Minghim and Haim Levkowitz

73

Tutorial: Visual Mining of Text Collections

3.4 Visual representations: graphs, surfaces, volumes, triangulations



© Copyright 2007 Rosane Minghim and Haim Levkowitz

74

Tutorial: Visual Mining of Text Collections

4. From Visualization To Visual Text Mining

- 4.1 Visualization Techniques for Multidimensional Data
- 4.2 Visualization techniques and systems for handling document collections
- 4.3 Visual Text Mining

© Copyright 2007 Rosane Minghim and Haim Levkowitz

75

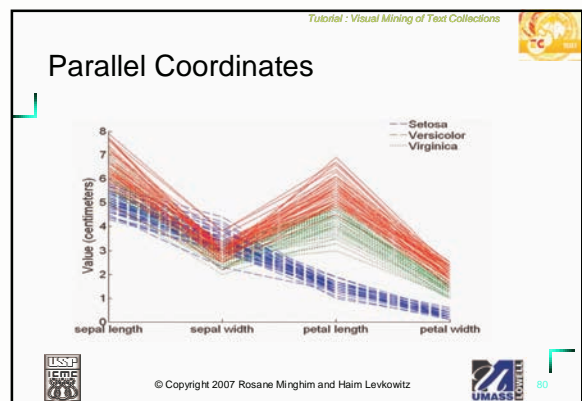
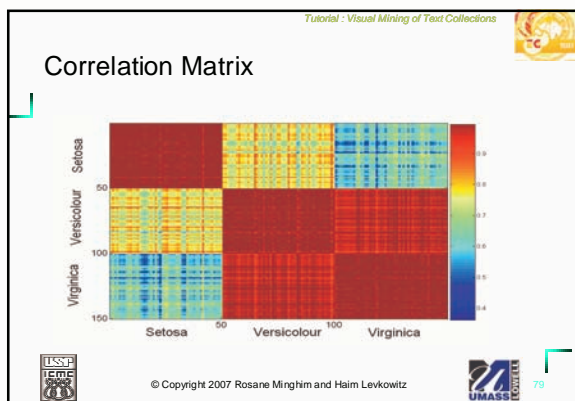
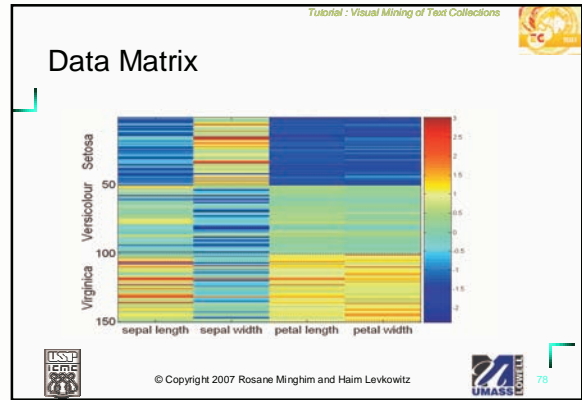
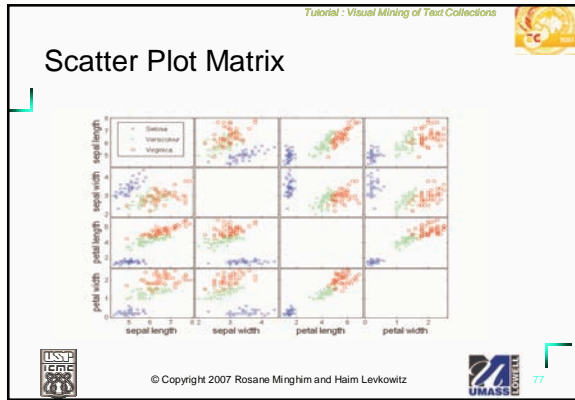
Tutorial: Visual Mining of Text Collections

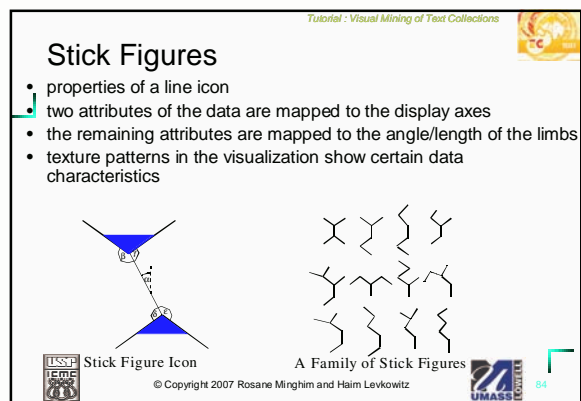
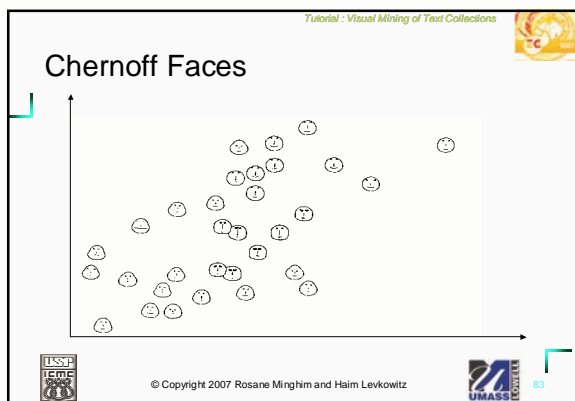
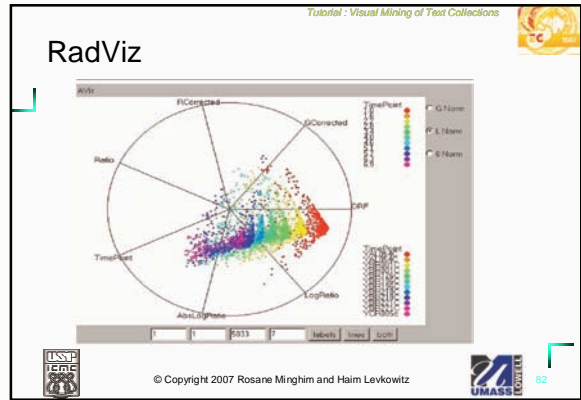
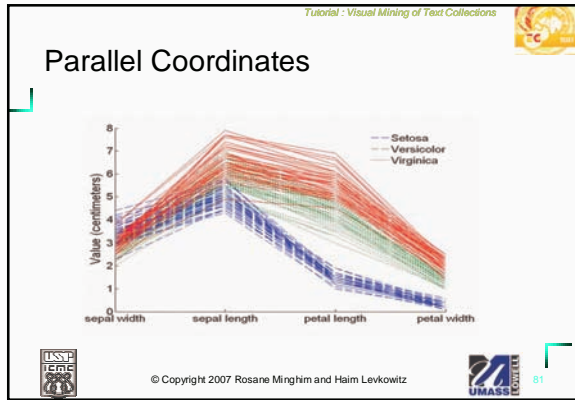
4. From Visualization To Visual Text Mining

- 4.1 Visualization Techniques for Multidimensional Data

© Copyright 2007 Rosane Minghim and Haim Levkowitz

76






Tutorial: Visual Mining of Text Collections

Stick Figure Icon

5-dimensional NOAA image data from the great lake region

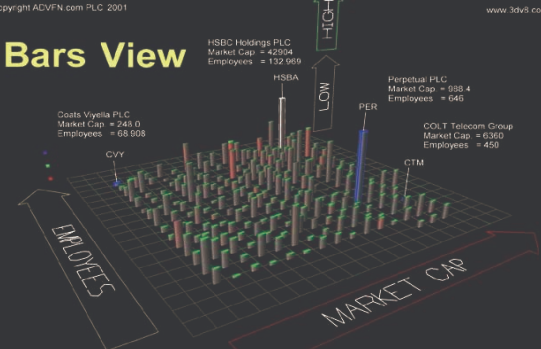


© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Bars View

Copyright ADVFN.com PLC 2001

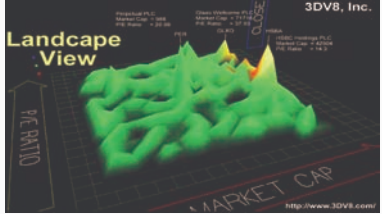


| Company | Market Cap | Employees |
|--------------------|------------|-----------|
| HSBC Holdings PLC | 42,004 | 132,969 |
| Goats Vynella PLC | 248.0 | 68,608 |
| PER | 588.4 | 646 |
| COIT Telecom Group | 6,356 | 450 |

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Landscape View



© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Pixel-oriented Techniques

- use each pixel to visualize one data value
 - about 1.3 million data values can thus be displayed at one time
- Need to map each data value to a color
- Thus need a color map and an interpretation for the map
- Also called dense pixel displays

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Dense Pixel Displays

- each attribute value is represented by one colored pixel (the value ranges of the attributes are mapped to a fixed color map)
- the attribute values for each attribute are presented in separate but connected sub-windows – splits apart the shape coding

visualization of six-dim. data

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Three Questions

How should the *pixels be arranged* within the subwindows?

Are alternative *shapes* of the sub-windows possible?

What is an appropriate ordering of the dimensions?

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Arrangement of Pixels

Given: Ordered Set of n data items $\{a_1, \dots, a_n\}$ consisting of k data values each (a_1^1, \dots, a_1^k)

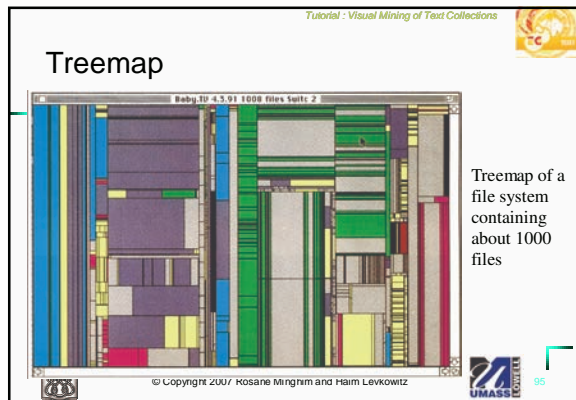
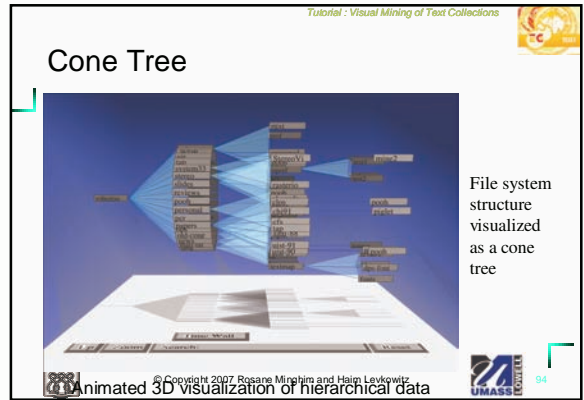
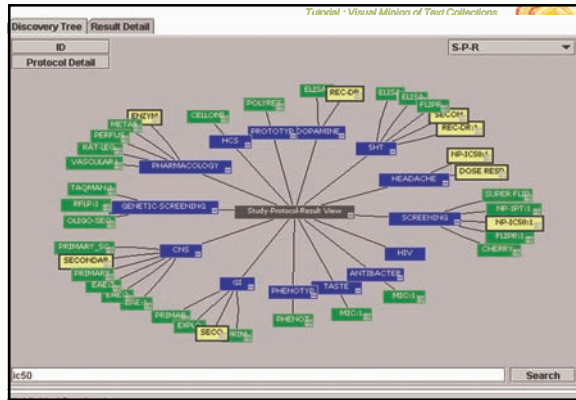
Goal: Two-dimensional arrangement of the data values, i.e. bijective mapping $f: \{1 \dots n\} \rightarrow \{1 \dots b\} \times \{1 \dots h\}$ ($n \leq b * h$), such that the function

$$\sum_{i=1}^n \sum_{j=1}^n \left| d(f(i), f(j)) - d\left((0,0), \left(b \cdot \sqrt{\frac{|i-j|}{n}}, h \cdot \sqrt{\frac{|i-j|}{n}} \right) \right) \right|$$

is minimal, where $d(f(i), f(j))$ is the L^p -distance ($p = 1, 2$) of the pixels belonging to a_i and a_j

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Space-Filling Curves




4. From Visualization To Visual Text Mining
- 4.2 Visualization techniques and systems for handling document collections
 - 4.3 Visual Text Mining
- © Copyright 2007, Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

IN-SPIRE

- Spatial Paradigm for Information Retrieval - Pacific Northwest National Laboratories
- Two Visualization Metaphors:
 - Galaxies – dimensional reduction
 - Themescape

© Copyright 2007 Rosane Minghim and Haim Levkowitz

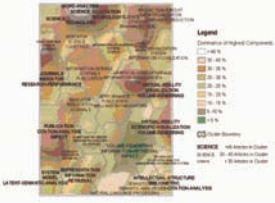


97


Tutorial: Visual Mining of Text Collections

SOM based

- Self-Organization Maps (SOMs) cartográficos (ex. Skurpin 2002)



© Copyright 2007 Rosane Minghim and Haim Levkowitz

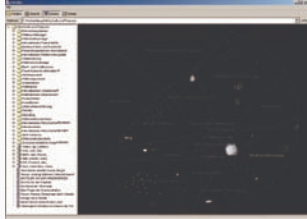


98

Tutorial: Visual Mining of Text Collections

InfoSky

Granitzer (Granitzer et al., 2004) also employs galaxy metaphor



© Copyright 2007 Rosane Minghim and Haim Levkowitz

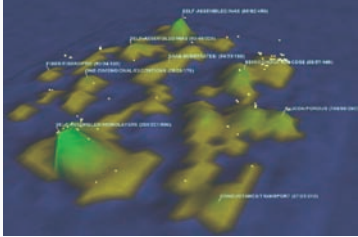


99


Tutorial: Visual Mining of Text Collections

VxInsight

- Sandia National Laboratories, mountain metaphor (Boyack et al., 2002).



© Copyright 2007 Rosane Minghim and Haim Levkowitz




100

Tutorial - Visual Mining of Text Collections

HIVE (Ross and Chalmers 2003)

- Interconnected components:
 - Import
 - Transform
 - Render multi-dim data



© Copyright 2007 Rosane Minghim and Haim Levkowitz


UMASS LOWELL

101

Tutorial - Visual Mining of Text Collections

Projection Explorer (PEX)

- Projection and Point placement
- Precision
- Graphs and surfaces (Super Spider)



© Copyright 2007 Rosane Minghim and Haim Levkowitz

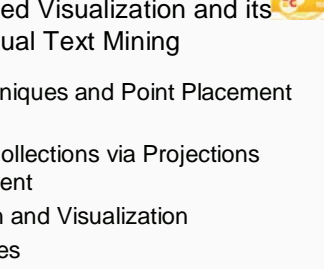
UMASS LOWELL

102

Tutorial - Visual Mining of Text Collections

5. Projection Based Visualization and its application to Visual Text Mining

- 5.1 Projection Techniques and Point Placement Strategies
- 5.2 Mapping Text Collections via Projections and Point Placement
- 5.3 Topic Extraction and Visualization
- 5.4 Further Examples



© Copyright 2007 Rosane Minghim and Haim Levkowitz

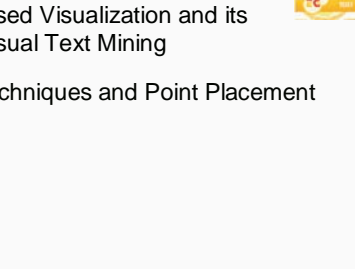
UMASS LOWELL

103

Tutorial - Visual Mining of Text Collections

5. Projection Based Visualization and its application to Visual Text Mining

- 5.1 Projection Techniques and Point Placement Strategies



© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL

104

Tutorial: Visual Mining of Text Collections

Ex: Sammon Mapping

- Let X be the points in the original space R^n , we apply a distance measure d_{ij}^* between X_i and X_j , and find Y , the **projected point**, ex. R^2 and d_{ij} the Euclidean distance between them.
- Sammon's method applies an error function to measure the target.

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Least-Square Projection (LSP)

- $S = \{p_1, p_2, \dots, p_n\} \in R^m$
- Preserving as much as possible the neighborhood relationship amongst points
- The neighborhood is defined by choosing the k -nearest neighbors of a point
- Least square Meshes

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Least-Square Projection (LSP)

- Least square Meshes
- Three main steps
 - Select a subset of S (*control points*) and project it onto R^d
 - Determine the neighborhood of the points
 - Build a linear system whose solutions are the Cartesian coordinates of the points p_i in R^d

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

Building the Linear System: Laplacian Matrix

- Let $V_i = \{p_{i1}, \dots, p_{ik}\}$ be a set of k_i points in a neighborhood of a point p_i and p_{ij} be the coordinates of p_{ij} in R^d

- each p_i is the centroid of the points in V_i

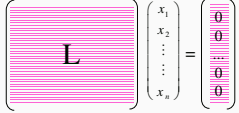
© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Building the Linear System: Laplacian Matrix

$Lx_1=0, Lx_2=0, \dots, Lx_d=0$

where x_1, x_2, \dots, x_d are the vectors containing the Cartesian coordinates of the points and L is the matrix given by

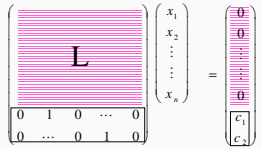
$$L_{ij} = \begin{cases} 1 & i = j \\ -\frac{1}{k_i} & p_j \in V_i \\ 0 & \text{otherwise} \end{cases}$$


© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Building the Linear System: Adding Control Points

$$C_{ij} = \begin{cases} 1 & p_j \text{ is a control point} \\ 0 & \text{otherwise} \end{cases}$$

$$b_i = \begin{cases} 0 & i \leq n \\ x_{p_{c_i}} & n < i \leq n + nc \end{cases}$$


© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Solving the System

- The system is solved in a least-square sense
- The unique analytical solution is $\|Ax - b\|^2$

$$A^T A x = A^T b \Rightarrow x = (A^T A)^{-1} A^T b$$

- is symmetric and sparse and can be solve using *Cholesky* factorization

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial : Visual Mining of Text Collections

Choosing the Control Points

- In order to select the control points
 - the space R^m is split into nc clusters using k-medoids.
 - the control points are the medoids of each cluster

© Copyright 2007 Rosane Minghim and Haim Levkowitz


Tutorial: Visual Mining of Text Collections

Choosing the Control Points

- Once the control points are chosen, these points are projected onto R^d through a fast dimensionality reduction method
 - Fastmap or NNP [2]
 - Further improvement of the projection using FORCE [2]

[2] E. Tejada, R. Minghim, and L. Nonato. On improved projection techniques to support visual exploration of multidimensional data sets. *Information Visualization Journal*, 2(4):218–231, 2003.

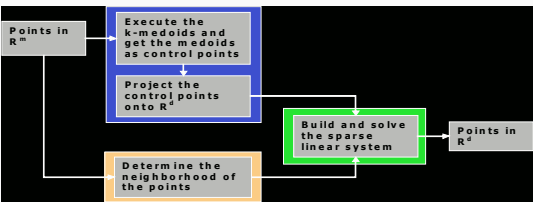
© Copyright 2007 Rosane Minghim and Haim Levkowitz




113

Tutorial: Visual Mining of Text Collections

LSP: Overview

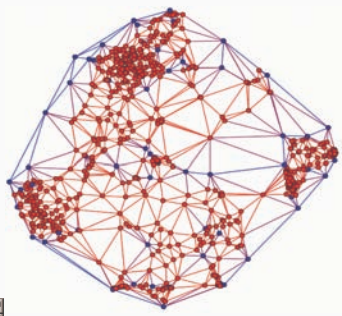


© Copyright 2007 Rosane Minghim and Haim Levkowitz




114

Tutorial: Visual Mining of Text Collections



Control points in blue

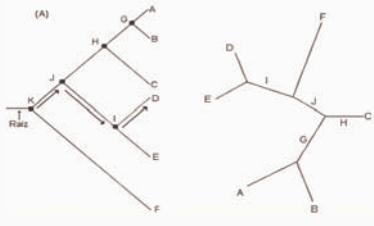
© Copyright 2007 Rosane Minghim and Haim Levkowitz




115

Tutorial: Visual Mining of Text Collections

Point Placement by Phylogenetic Tree Construction Algorithms (N-J Trees)



© Copyright 2007 Rosane Minghim and Haim Levkowitz



116

Tutorial - Visual Mining of Text Collections

Point Placement by Phylogenetic Tree Construction Algorithms (N-J Trees)

$$d_{AB} + d_{CD} \leq \max(d_{AC} + d_{BD}, d_{AD} + d_{BC})$$

$d_{AB} = a + b$ $d_{AC} = a + e + c$ $d_{AD} = a + e + d$
 $d_{CD} = c + d$ $d_{BD} = b + e + d$ $d_{BC} = b + e + a$

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial - Visual Mining of Text Collections

Algorithm Neighbor-joining

Input: distance matrix

1. Create a star tree for n objects.
2. Iteration
 1. Select a node pair (i,j) with smaller S_{ij} (branch size)

$$S_{ij} = \frac{1}{2(n-2)} \sum_{k=3}^n (D_{ik} + D_{jk}) + \frac{1}{2} D_{ij} + \frac{1}{n-2} \sum_{k=3, k \neq i, j}^n D_{ik}$$
 2. Combine nodes i and j in a new node and calculate the branch size of the new node.

$$L_{ik} = \frac{D_{ij} + D_{iz} - D_{jz}}{2} \qquad L_{jk} = \frac{D_{ij} + D_{jz} - D_{iz}}{2}$$

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial - Visual Mining of Text Collections

Algorithm Neighbor-joining

3. Calculate new distance matrix, computing the new distances from the new node to the remaining nodes.

$$D_{(i-j),k} = \frac{(D_{ik} + D_{jk})}{2} \quad (3 \leq k \leq N)$$
4. Eliminate previous nodes i and j
5. If $n > 2$ then iterate again.

© Copyright 2007 Rosane Minghim and Haim Levkowitz


Tutorial - Visual Mining of Text Collections

© Copyright 2007 Rosane Minghim and Haim Levkowitz


Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

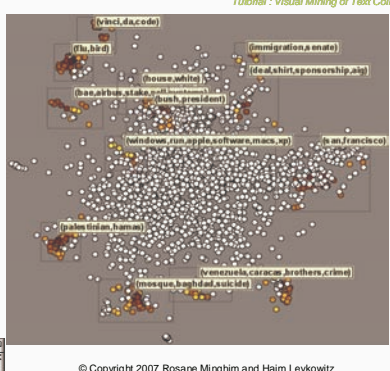
- Positioning and labeling




© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial: Visual Mining of Text Collections




© Copyright 2007 Rosane Minghim and Haim Levkowitz




Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

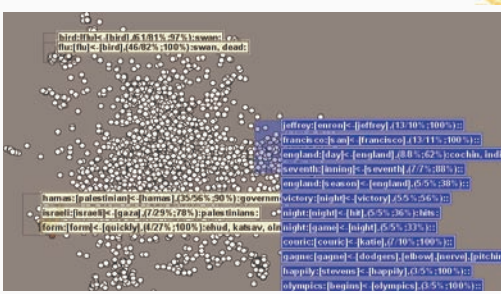
- Detailing topics




© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial: Visual Mining of Text Collections




© Copyright 2007 Rosane Minghim and Haim Levkowitz




Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

- Finding Relationships

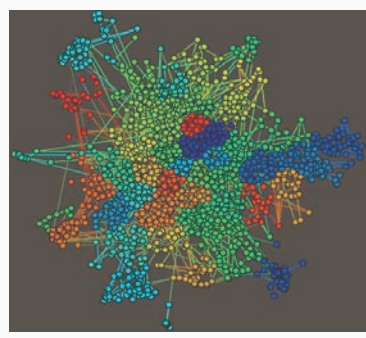


© Copyright 2007 Rosane Minghim and Haim Levkowitz




125

Tutorial: Visual Mining of Text Collections



© Copyright 2007 Rosane Minghim and Haim Levkowitz




126


Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

- Untangling

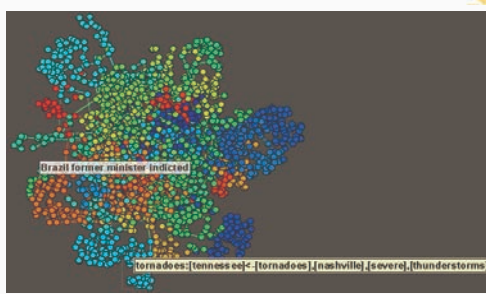


© Copyright 2007 Rosane Minghim and Haim Levkowitz




127

Tutorial: Visual Mining of Text Collections



© Copyright 2007 Rosane Minghim and Haim Levkowitz




128

Tutorial : Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

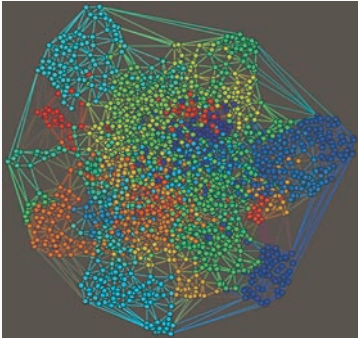
- Building a Surface - meshing



© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 129

Tutorial : Visual Mining of Text Collections




© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 130

Tutorial : Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

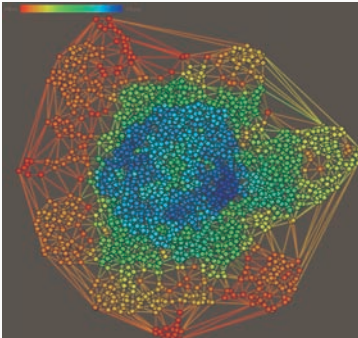
- Coloring by degree of proximity



© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 131

Tutorial : Visual Mining of Text Collections



© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 132

Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

- Alternate view (N-J Tree)

→

USSP UMass Lowell 133

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

USSP UMass Lowell 134

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

- Coordinating

→

USSP UMass Lowell 135

© Copyright 2007 Rosane Minghim and Haim Levkowitz

Tutorial: Visual Mining of Text Collections

USSP UMass Lowell 136

© Copyright 2007 Rosane Minghim and Haim Levkowitz


Tutorial : Visual Mining of Text Collections

5.2 Mapping Text Collections via Projections and Point Placement

- Building a Surface

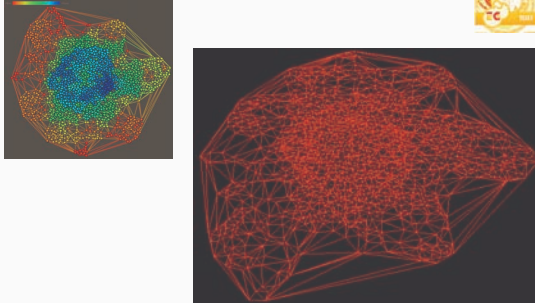
→

© Copyright 2007 Rosane Minghim and Haim Levkowitz




137

Tutorial : Visual Mining of Text Collections

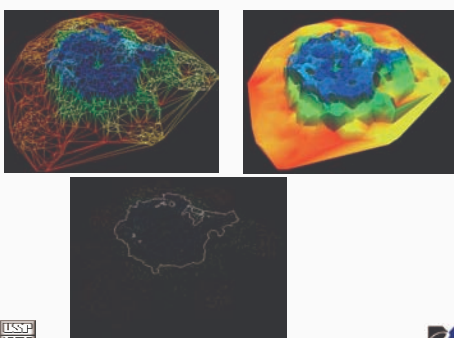


© Copyright 2007 Rosane Minghim and Haim Levkowitz




138

Tutorial : Visual Mining of Text Collections



© Copyright 2007 Rosane Minghim and Haim Levkowitz




139

Tutorial : Visual Mining of Text Collections

5.3 Topic Extraction and Visualization

- Topic Definition by Covariance
- Topic Extraction by Seeded Generation of Association Rules (pruning by relevant terms)
- Labeling and Viewing

© Copyright 2007 Rosane Minghim and Haim Levkowitz



140

Tutorial - Visual Mining of Text Collections


Topic Extraction and Visualization

Topic Definition by Covariance


- Pair of words with highest covariance

$$cov(t_i, t_j) = \frac{1}{n-1} \sum_{k=1}^n (t_{ki} - \bar{t}_i)(t_{kj} - \bar{t}_j)$$

- For all the other words, highest mean covariance compared to first two.
- Add to label if above threshold.

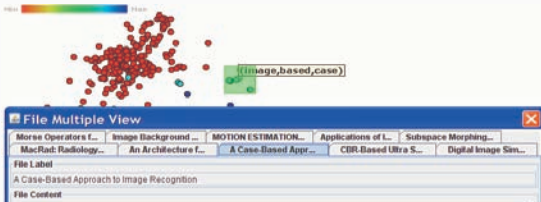



© Copyright 2007 Rosane Minghim and Haim Levkowitz


141


Tutorial - Visual Mining of Text Collections

Topic Definition by Covariance





© Copyright 2007 Rosane Minghim and Haim Levkowitz



142

Tutorial - Visual Mining of Text Collections


Topic Extraction and Visualization

Topic Extraction using Association Rules

- Use relevant words as seeds
- Prune the case by rule weighting



© Copyright 2007 Rosane Minghim and Haim Levkowitz



143

Tutorial - Visual Mining of Text Collections


Topic Extraction using Association Rules

| Transactions | Items | Frequent Itemsets | Support |
|--------------|-----------------------------|---------------------|---------|
| 1 | Trousers, t-shirt, snickers | {snickers} | 75% |
| 2 | T-shirt, snickers | {Trousers} | 50% |
| 3 | shorts, snickers | {T-shirt} | 50% |
| 4 | Trousers, sandals | {T-shirt, snickers} | 50% |

Min. support = 50% (2 transactions).
Min. confidence = 50%.



© Copyright 2007 Rosane Minghim and Haim Levkowitz


144

Tutorial: Visual Mining of Text Collections

Topic Extraction using Association Rules

tenis → t-shirt

$$\text{support} = \text{support}(\{\text{snickers}, t\text{-shirt}\}) = 50\%$$


$$\text{confidence} = \frac{\text{support}(\{\text{snickers}, t\text{-shirt}\})}{\text{support}(t\text{-shirt})} = \frac{50}{50} = 100\%$$

T-shirt → snickers

$$\text{support} = \text{support}(\{\text{snickers}, t\text{-shirt}\}) = 50\%$$

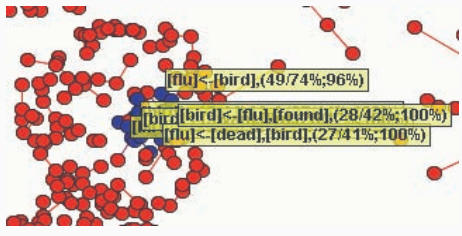
$$\text{confidence} = \frac{\text{support}(\{\text{snickers}, t\text{-shirt}\})}{\text{support}(\text{snickers})} = \frac{50}{75} = 66,6\%$$

© Copyright 2007 Rosane Minghim and Haim Levkowitz




Tutorial: Visual Mining of Text Collections

Topic Extraction using Association Rules (example)

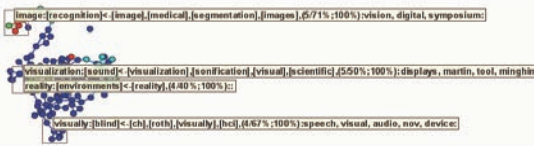


© Copyright 2007 Rosane Minghim and Haim Levkowitz




Tutorial: Visual Mining of Text Collections

Topic Extraction using Association Rules (example)



© Copyright 2007 Rosane Minghim and Haim Levkowitz




Tutorial: Visual Mining of Text Collections

Topic Extraction using Association Rules

- Topics using AR
 - Term co-occurrence in documents \Leftrightarrow subject
 - Transaction \Rightarrow Document
 - Item \Rightarrow term

© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial : Visual Mining of Text Collections

Topic Extraction using Association Rules

- Issues
 - Discovered rules amount
 - Term relevance (items)
 - Rule relevance measure (filtering)
- High Sup. & conf. => few interesting rules
- Low Sup. & conf. => huge amount of rules

UMASS LOWELL

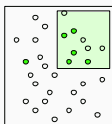
© Copyright 2007 Rosane Minghim and Haim Levkowitz

149

Tutorial : Visual Mining of Text Collections

Locally weighted and seeded AR

- Weighting Terms and Rules



$$w_{i,j} = \frac{\sum_{j=1}^k Tf_{i,j}}{\sum_{j=1}^k Tf_i}$$

$w_{i,s} = 5/6 = 0.83333$

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

150

Tutorial : Visual Mining of Text Collections

Steps

1. S: set of user selected documents
2. Picked 10 most relevant terms

$$W_{t_j S_k} = \frac{\sum Tf_{t_j S_k}}{\sum Tf_{t_j C}}$$

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz

151

Tutorial : Visual Mining of Text Collections

Steps

1. Initial item sets: Tr x T
 - Relevant Terms x All Terms
2. Items Sets discovered by Apriori algorithm
3. Sorted by weight:

$$\sum W_{t_j S_k}$$

UMASS LOWELL

© Copyright 2007 Rosane Minghim and Haim Levkowitz


152

Tutorial - Visual Mining of Text Collections

Steps

6. Highest weight item set selected
7. Covered documents removed from S
8. Further item sets are selected if there is support over residual S (repeats 6 e 7)
9. If all items sets are considered and $|S \text{ residual}| > 0$, repeats whole process with residual S.

© Copyright 2007 Rosane Minghim and Haim Levkowitz




153

Tutorial - Visual Mining of Text Collections

Sequential covering with Multiple restart

- Variance and Coverage
- Partitioning Strategies
- Grid
 - Resize
 - Slide
- Cluster
 - Cluster number

© Copyright 2007 Rosane Minghim and Haim Levkowitz




154

Tutorial - Visual Mining of Text Collections

5.4 Further Examples

- RSS Patent Data, recovered from the Web <http://www.freepatentsonline.com/>
- Case 1:
 - 170 files
 - Graphics processing, printer, database, document, ai

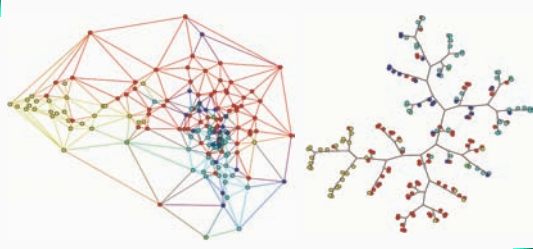
© Copyright 2007 Rosane Minghim and Haim Levkowitz




155

Tutorial - Visual Mining of Text Collections

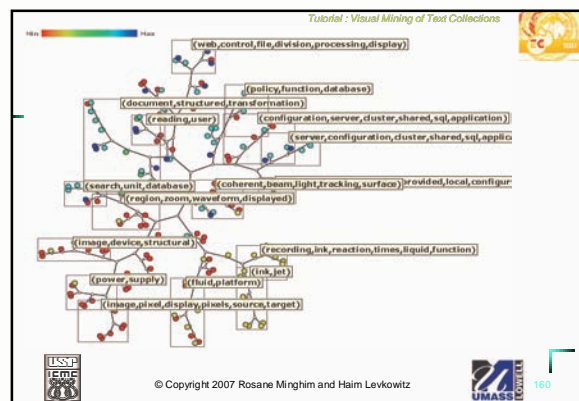
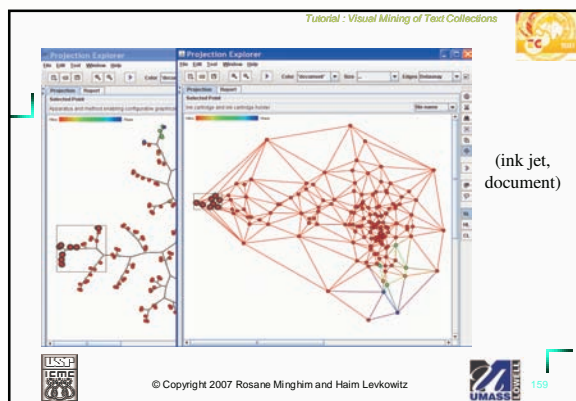
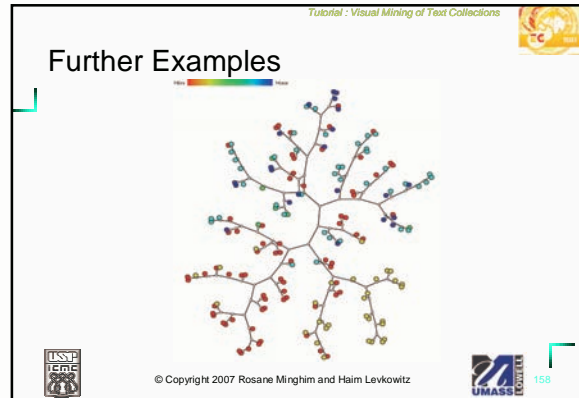
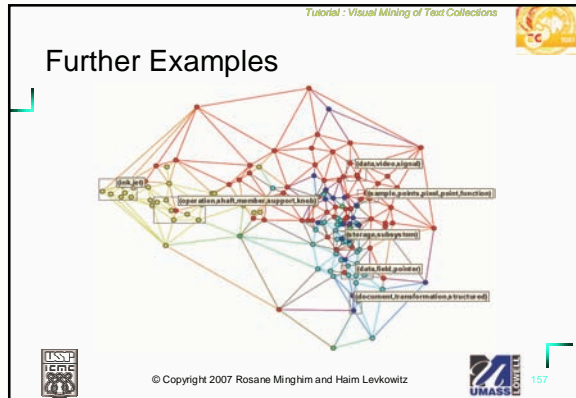
Further Examples

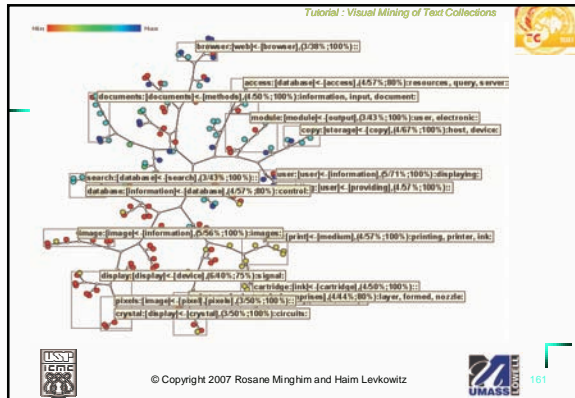


© Copyright 2007 Rosane Minghim and Haim Levkowitz



156



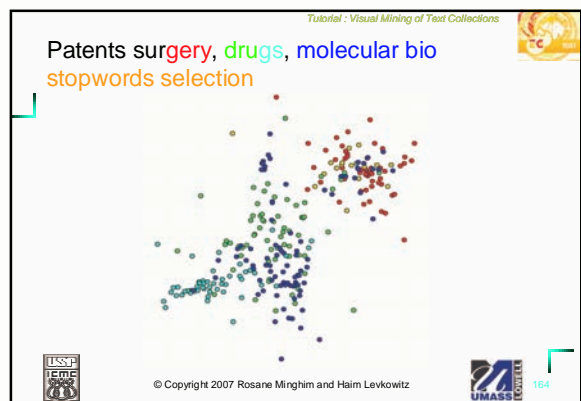
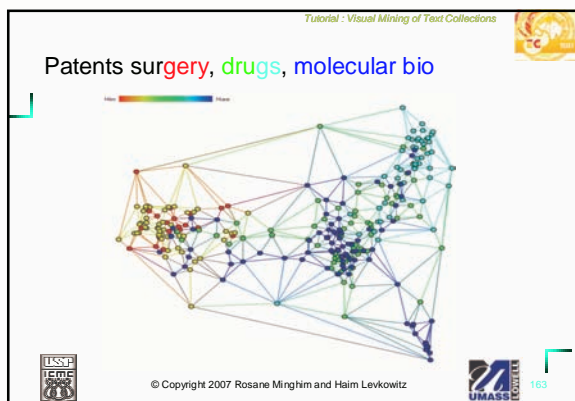


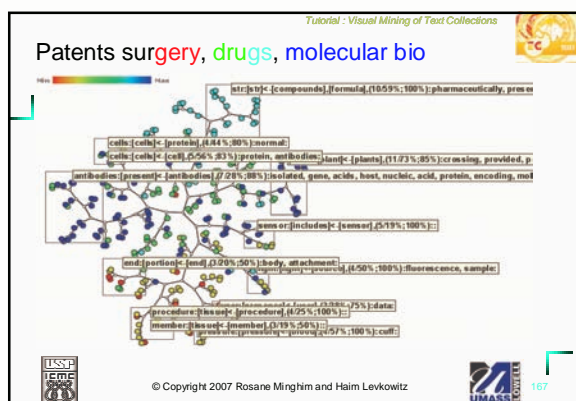
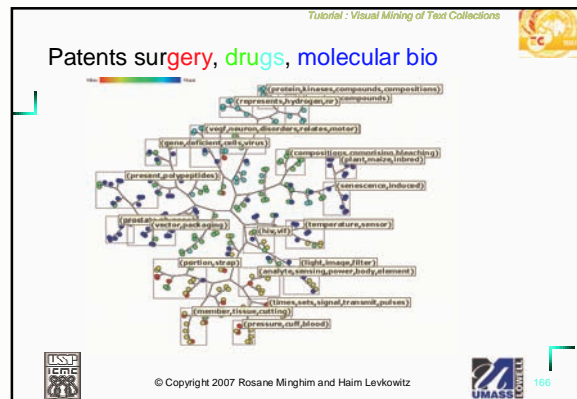
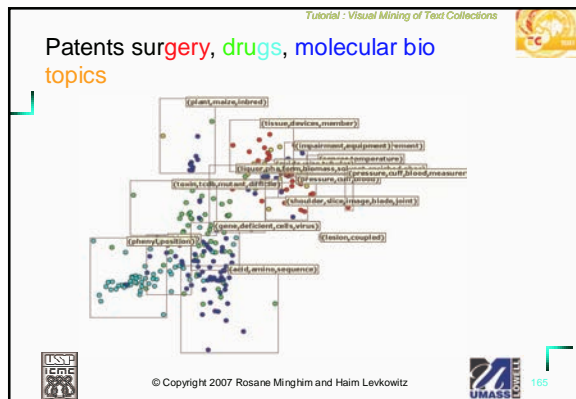
Tutorial: Visual Mining of Text Collections

Patents – case 2

- <http://www.freepatentsonline.com/>
- 172 files
- surgery (2), drugs(2), molecular biology

© Copyright 2007 Rosane Minghim and Haim Levkowitz





Tutorial: Visual Mining of Text Collections

Further Examples

- Cattle performance data
 - Translated to text from categorical information, e.g.,
 - Ranges of weight to words such as:
 - {weight_below_fifty_percent;
 - weight_between_fifty_seventy_five; etc..}
 - 9135 individuals

© Copyright 2007 Rosane Minghim and Haim Levkowitz



Tutorial : Visual Mining of Text Collections

6. Conclusions, Challenges, Trends

- Other M-D data
- Measures of similarity
- Metrics for quality of projections and point placement
- Interaction and Exploration Paradigms
- Time representation

© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 170

Tutorial : Visual Mining of Text Collections

Conclusions, Challenges, Trends

- Vector Representation
 - Stopword lists/vocabulary
 - Alternative representations
 - Heterogeneous vs. Homogeneous
- Scalability: two levels?
 - Massive
 - Close examination

© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 171

Tutorial : Visual Mining of Text Collections

Conclusions, Challenges, Trends

- Evaluation
- Customization
- Further coupling with mining
 - Supervised/non-supervised
- Coordination, Multiple Views

© Copyright 2007 Rosane Minghim and Haim Levkowitz

UMASS LOWELL 172