A Cartographic Approach to Visualizing Conference Abstracts

André Skupin
University of New Orleans

A cartographic approach to mapping nongeographic information helps to manage graphic complexity in visualizations. It aids domain comprehension by forcing us to use the same cognitive skills we use when viewing geographic maps.

The map metaphor has become popular in recent works on information visualization. However, as we attempt to visualize information spaces in a low-dimensional display space, crucial impediments to the metaphor's usefulness appear: projection techniques break down because of a lack of scalability, visualizations suffer from graphic complexity, and labels are imbued with too little interpretable meaning and are hard to position without conflict.[1] Cartographers have traditionally addressed many of these problems, albeit in a strictly geographic context. They've found numerous ways to represent a given portion of the infinitely complex earth surface on a finite map surface, be it on paper or on a computer screen.

With this in mind, I present a distinctly cartographic approach to mapping nongeographic information. Focusing on the text content of a set of conference abstracts, we can derive 2D visualizations of information spaces that address complexity and automation.

Cartographically informed abstraction

Apart from the depiction of geographic space itself, information visualizations rarely make use of cartographic principles, particularly with respect to issues of graphic complexity. There are hopeful signs, however, that this might be changing as evidenced by Foley's list of "The Ten Top Problems Left" in computer graphics:

> When we want to create an abstraction that conveys key ideas while suppressing irrelevant detail, we need to draw on … the vast knowledge of cartographers and animators.[2]

Cartographers and geographers are now involved in nongeographic information visualization in several ways. Combined with the influence of cognitive linguists, the desire to extend certain geographic notions and principles to nongeographic information drives much of this work.[3,4] This refers particularly to questions regarding the nature of geographic space, the objects that inhabit it, and the ways in which humans conceptualize it.

In an earlier paper, I discussed a broad range of geographic considerations and cartographic techniques as they relate to the visualization of text documents.[1] Here, I show how we can further improve map-like visualizations of nongeographic information. The principal approach and specific techniques I use relate to noted document visualization efforts such as ThemeScapes,[5] ET-Map,[6] Depict,[7] and WebSOM.[8] I attempt to create visualizations that subjectively look like maps, thereby forcing the audience to use the same cognitive skills typically associated with geographic maps.

The experiment that I describe in this article is based on a set of 2,220 abstracts submitted to the Annual Meeting of the Association of American Geographers (AAG), held in Honolulu, Hawaii, in March 1999. The whole range of the geography discipline is represented at the annual meeting. Ideally, a visual representation of the corpus of abstracts should thus paint a fairly comprehensive picture of the current state of the field. Information visualization should also convey valuable insight into the status and semantic relationships of the various research interests represented by the AAG's 50 or so specialty groups.

My method is informed by the way in which we derive topographic maps and some thematic maps. First, we create a detailed base map in which each element of an information space occupies a discrete 2D position. As the available display surface decreases—cartographers refer to this as *scale reduction*—generalized versions are created so that graphic density is reduced and high-level structures of the depicted space are brought to the fore.

0272-1716/02/$17.00 © 2002 IEEE

In traditional cartography, this is done through a combination of geometric and conceptual operations. In this research I produce generalized versions by merging individual features into groups through hierarchical clustering, based on feature similarity. Since the computation of the original base map is also driven by feature similarity, the high-dimensional merging of features manifests itself as a merging of geometric elements in the 2D display surface. Differences between various levels of abstraction are reflected not only in changing geometric configurations but also in the text labels associated with individual and clustered features. Rich, scale-dependent labeling helps observers make sense of the many semantic facets inherent in a document corpus.

Base map creation

Creation of the base map is driven by the desire to express document similarity through geometric proximity. At the same time, the computational procedures should be scalable toward larger document collections and result in a geometric structure that easily supports feature aggregation across large-scale ranges. The result is a methodology for transforming a document corpus into a tessellated configuration in which each document corresponds to a unique polygon.

Keyword index and vector-space model

Content-based visualization of text documents typically starts with creating a keyword index. This remains one of the most problematic issues, especially if an index is to be created fully automatically, without prior knowledge of the particular domain and if entries are to be restricted to the kind of meaningful terms that human indexers would choose.

The procedure that I describe allows the fully automated creation of an index while restricting the keyword set to relatively meaningful terms. However, it's restricted to information spaces in which the vast majority of documents have author-chosen keywords associated with them. Conference abstracts fall into this category. Authors of conference abstracts are usually asked to provide between three and five keywords.

While these keywords are a prerequisite of the indexing process, I use them here in a manner that differs from other approaches. Instead of forming the index itself, I broke the author-chosen keywords into single words. For example, I'd break an original keyword of "glacial geomorphology" into the components "glacial" and "geomorphology." The exclusive use of keyword components chosen by the authors ensures that a meaningful index can later be created without major processing effort and without resorting to more advanced methods aimed at the extraction of meaning-bearing terms. Although I used some stemming (that is, I reduced words to such stems as "geogr" or "geol"), I retained the original terms for the most part, which admittedly leads to some duplication of meaning when such words as "forest" and "forests" are treated as separate terms.

I then matched keyword components against the full text of all abstracts to create the actual index. For each abstract, I recorded whether and how many times it contains a certain keyword component. This approach has two major advantages over the exclusive use of author-chosen keywords for each abstract. First, it allows an automatic indexing of those abstracts that aren't accompanied by any keywords, as long as most authors add keywords to their abstracts. (Authors of 174 abstracts, or 7.8 percent, didn't submit any keywords.) Second, my approach leads to a richer vector-space model that allows more differentiation when comparing the articles' content. For example, two abstracts sharing both "glacial" and "geomorphology" would express more similarity than two abstracts that have only one of those keywords in common.

To express the relationship between a set of keywords and a document corpus, we can use the vector-space model approach widely publicized by the work of Gerard Salton. In my experiment, I created a term-document matrix filled with the raw keyword counts for each document. A vector of term counts thus takes the place of the full-text document for all further processing.[9]

Self-organizing maps for document visualization

Self-organizing maps (SOMs), also known as Kohonen maps, are commonly used to process the kind of high-dimensional vector space model presented here (see the sidebar "Self-Organizing Maps", next page). Use of the SOM method for document visualization was first demonstrated 10 years ago.[10] The scalability of the technique for large data sets and the attractiveness of the resulting visualizations have continued to spark the interest of numerous researchers working on document visualization.[6-8]

Input to the SOM method consists of a matrix containing objects (rows) and their respective attribute values (columns). When applied to document spaces, this can correspond to a term-document matrix, as I discussed earlier. The number of terms associated with documents can vary widely. In the experiment's data set it ranged from three to 57 keywords (Figure 1, next page). This can have unintended consequences whenever we use Euclidean and similar metric measures, as is typically the case for SOM algorithms. Documents with few keywords will tend to be drawn together while those with many keywords will be pushed apart, despite actual keyword matches that might occur between short and long documents. After testing a number of normalization schemes, I addressed this problem by filtering the keyword index around the mean keyword count. I removed 1,052 documents containing less than 18 or more than 28 keywords from the term-document matrix prior to SOM training, leaving 1,148 documents.

The training phase is the most time-consuming portion of the SOM method. Although users choose the number of iterations, they usually range from several thousand for small SOMs to several hundred thousand for very high-dimensional data sets. The result of the learning process is a 2D, raster-like representation of n-dimensional term space, in which raster elements correspond to individual neurons.

The time it takes to train a SOM and the way in which it can be used depend on its size, or the number of nodes it has. We can train a small SOM much faster, which

Self-Organizing Maps

The self-organizing map (SOM) method is one of the more widely used forms of artificial neural networks (ANN). Input to the technique is a set of *n*-dimensional observations. The output layer consists of a network of neurons or nodes, which are typically arranged as a 2D lattice, forming a configuration akin to raster data models. Nodes are connected to their neighbors according to either a square or hexagonal neighborhood definition (see Figures 2 and 3a). Contrary to other ANN approaches, there's no hidden layer. Associated with each neuron is a reference vector of the same dimensionality as the input data. Input vectors train the neuron grid so that topological relationships among input observations are preserved. Adjustments to the reference vector of a particular neuron aren't done in isolation but propagate to its network neighbors. Training a SOM can be time-consuming if the dimensionality of input vectors and/or the number of network nodes are high.

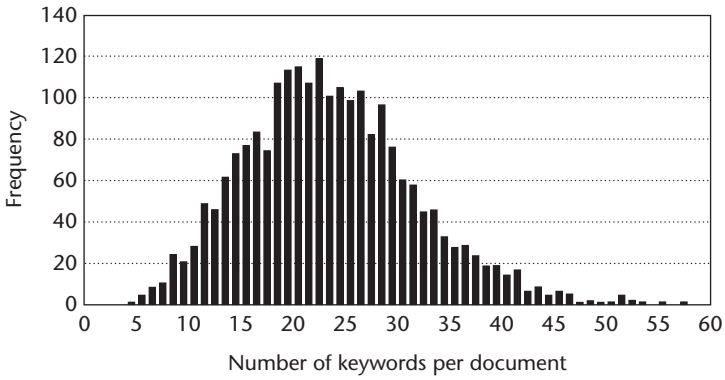
We can use the trained SOM in several ways. For example, we can visualize individual vector components to see areas of the map that are of particular significance. A popular form of visualization is known as the U-matrix method, in which a focal operator computes differences among neighboring reference vectors. The resulting patterns can be interpreted as clusters. We can also compute clusters more explicitly by applying standard clustering techniques to all the neurons. New observations can be mapped onto the trained SOM quickly, since it only involves finding the best matching reference vector. For example, in the case of conference abstracts, we could find out how our own research interests fit in with the rest of the conference program.

For an in-depth discussion of the SOM method, see Kohonen's book.[1] A number of collections of SOM applications and case studies are also available.[2,3]

I based the visualization of conference abstracts presented in this article on a SOM trained with SOM_PAK 3.1, which is freely available from the Laboratory of Computer and Information Science, Helsinki University of Technology (http://www.cis.hut.fi/research/som_pak/). Final visualization, including automated label placement, was done in ESRI ArcGIS 8.1 (http://www.esri.com/), since SOM_PAK is limited in terms of interactivity and graphic output. SOM software also comes in the form of standalone, commercial packages, like Viscovery SOMine (http://www.eudaptics.com), which includes various visualization options. Finally, SOM functionality is increasingly available in connection with existing statistical and mathematical software products. The Neural Network Toolbox is an example (http://www.mathworks.com).

References

1. T. Kohonen, *Self-Organizing Maps*, Springer, Berlin, 1995.
2. G. Deboeck and T. Kohonen, *Visual Explorations in Finance with Self-Organizing Maps*, Springer, London, 1998.
3. E. Oja and S. Kaski, eds., *Kohonen Maps*, Elsevier Science, Amsterdam, 1999.



1 Conference abstracts can have a wide range in the number of keyword components. Documents around the mean of this distribution are used to train the SOM.

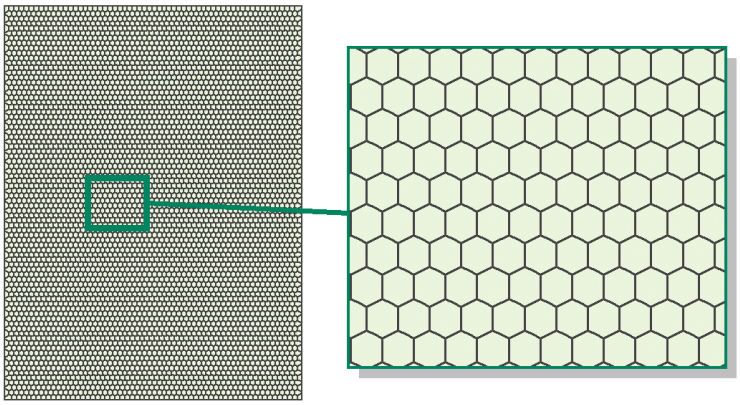
amounts to an unsupervised clustering of documents. For example, a SOM consisting of 4-by-4 nodes would divide documents into up to 16 clusters. This division of documents resembles *k*-means clustering, whose objective function is identical to Kohonen's algorithm. Many document visualizations involve the creation of relatively small SOMs. Lin trained a SOM of 140 nodes,[10] while the Depict system described by Rushall and Illgen had a default size of 400 nodes.[7]

Training a much finer SOM, particularly with high-dimensional data, can increase computing times by several orders of magnitude. At the extreme end there's the WebSOM project, which included the creation of a SOM for almost seven million patent abstracts. The WebSOM research group reports that multistep training of more than one million nodes took about six weeks on a six-processor system.[8]

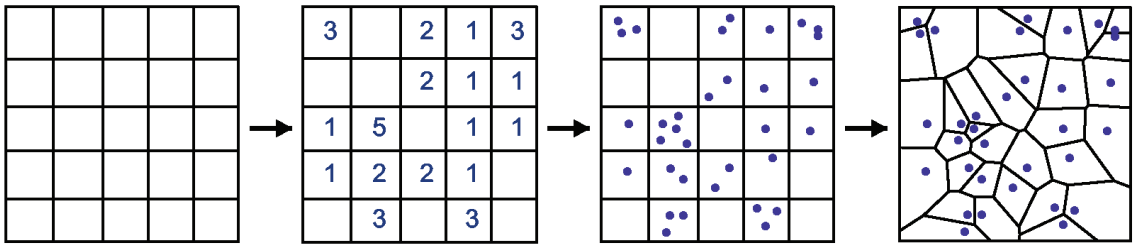
For this experiment, I trained a SOM of relatively fine resolution to create a base configuration in which the SOM recognizes and geometrically preserves many of the finer differences among abstracts. I trained a SOM consisting of 4,800 nodes (Figure 2) using the filtered index of 1,148 documents and 741 keyword components. The training stage alone, without subsequent clustering and visualization, took three hours on a Sun Ultra 1 workstation (200 MHz, 128 Mbytes), using SOM_PAK 3.1.

Visualization of individual documents

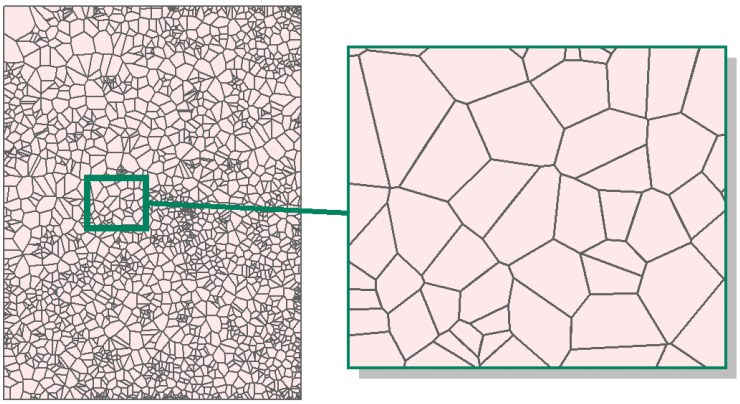
Once we establish the 2D configuration of neurons, it can help us determine the 2D positions of documents. Individual documents are assigned to the most similar neuron by comparing document vectors to neuron vectors. A single neuron may become associated with multiple documents. While this raster-type geometric configuration of neurons is typical for most SOMs, the goal in this experiment is to derive a set of discrete vector-type locations for each document. This allows direct access to single documents as well as to the attributes that may be associated with them, such as author name, modification date, and so on. Such a representation then serves as a kind of base map of the information, from which we can later derive generalized versions (analogous to the role of topographic base scales for the derivation of medium-scale topographic maps).



2 Geometric arrangement of 60×80 neurons for a self-organizing map. Each neuron's explicit connection with its six neighbors preserves topological relationships of the input data during training.



3 Deriving a point visualization from a self-organizing map and partitioning of map space into distinct polygons for each document. (a) Geometry of a square-based SOM. (b) Multiple documents (observations) may be associated with single neurons. Here, I indicate the number of observations per neuron. (c) Assignment of random locations inside neurons. (d) Complete tessellation of map space with Thiessen polygons.



4 Geometric base map configuration of conference abstracts.

The approach I propose here starts with a SOM of relatively fine resolution (see Figures 2 and 3a). After applying this SOM to a set of documents, neurons will have a varying number of documents associated with them (Figure 3b). I assign documents unique coordinate locations by randomly distributing them within the boundaries of the respective neuron grid cell (Figure 3c). This results in a geometric configuration containing regions of varying point density. To better enable the scale-dependent merging of individual documents into clusters, I assign points distinct portions of the 2D display area by using Thiessen or Voronoi polygons (Figure 3d). I applied this procedure to the complete set of conference abstracts. The result is a base map of the document corpus, consisting of 2,220 polygons (Figure 4).

Cluster-based generalization

When applied to a large set of documents, representations of the complete base map will quickly become too complex, especially if documents are to be accompanied by meaningful label terms. As a result, we need to devise methods to simplify the 2D base map so that content remains legible and meaningful at various scales. In this work, I simplified the visualization by merging neighboring document polygons if they were part of a statistically determined high-dimensional cluster. Whichever clustering method we adopt, we want to let viewers employ geographic notions of topology, proximity, clustering, and regionalization in the visualization of an information space.

Hierarchies in Scale-Dependent Visualization

Few approaches lend themselves to geometric and semantic abstraction like a hierarchical organization of information. Consider the administrative division of geographic space into countries, states or provinces, and counties, parishes, or districts. The nature of this structure as a nested hierarchy is reflected in the cartographic means used to visualize them, for example, by choosing symbols that convey a visual hierarchy. It's also standard practice in geographic information system (GIS) interfaces to define scale ranges within which certain layers will be either displayed or hidden.

Similarly, hierarchies can be great enablers for scale-dependent visualization of nongeographic information. The tree map method[1] was among the first to demonstrate this. A number of related products have found widespread use in recent years in such application areas as stock market visualizations (see http://www.smartmoney.com/maps). Formal investigation and computational modeling of scale dependency in information visualization are still rare, though. When it's done, as in the case of space-scale diagrams[2] or the zoomable user interface of Pad++,[3] similarities to the geographic treatment of scale in traditional cartography and modern GIS become obvious.

Hierarchical clustering

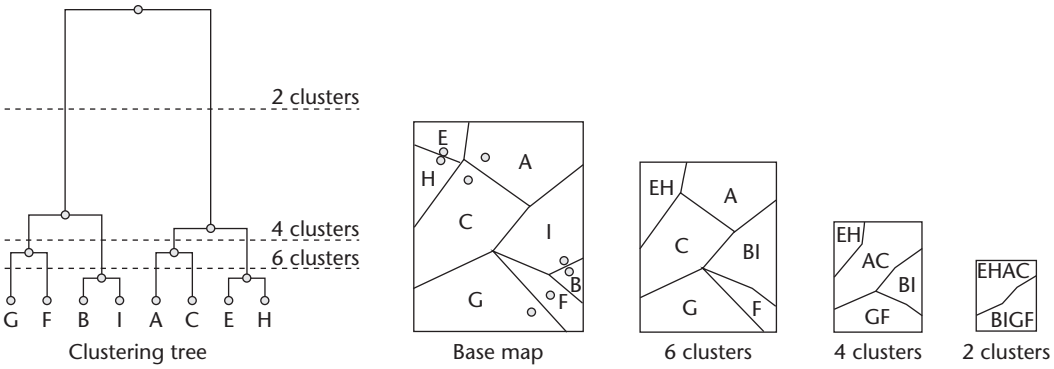
The choice of hierarchical clustering for a scale-dependent merging of individual features is natural. Many methods for computing clustering trees have been widely accepted and understood for several decades[4,5] and the respective hierarchical data structures are easy to build and maintain. Linking a clustering tree with a 2D base map allows control over the amount of detail shown at a given zoom level. The four maps shown in Figure A illustrate how we can manage graphic complexity in this manner. Base map polygons are derived from point locations as Voronoi diagrams. Each map corresponds to a certain level of aggregation in the clustering tree. In addition to such zoom-oriented interaction, we can also explore data by sliding up and down along the clustering tree and observing the merging or splitting of clusters in a map window of constant size.

Other hierarchies

Most zoomable information visualizations aren't based on the kind of computed hierarchies discussed in this article. They're typically based either on existing structural hierarchies or on constructed and managed content hierarchies. File system structures are an example for the former while Netscape's Open Directory Project (http://www.dmoz.org) exemplifies the latter. ODP data form the basis of a number of competing map-like Web visualization interfaces (compare the interface at http://www.webmap.com to the one at http://maps.map.net). What all of these approaches have in common is that the 2D location of features is primarily defined by their place in a nested hierarchy, not by interfeature relationships.

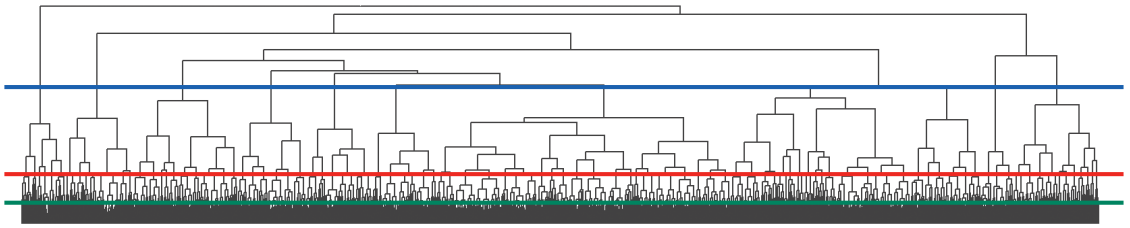
References

1. B. Johnson and B. Shneiderman, "Tree Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures," *Proc. IEEE Visualization 91*, IEEE CS Press, Los Alamitos, Calif., 1991, pp. 275-282.
2. G. Furnas and B. Bederson, "Space-Scale Diagrams: Understanding Multiscale Interfaces," *Proc. ACM Conf. Human Factors in Computing Systems* (CHI 95), ACM Press, New York, 1995, pp. 234-241.
3. B. Bederson and J. Hollan, "Pad++: A Zooming Interface for Exploring Alternative Interface Physics," *Proc. ACM Symp. User Interface Software and Technology* (UIST 94), ACM Press, New York, 1994, pp. 17-22.
4. P. Sneath and R. Sokal, *Numerical Taxonomy: The Principles and Practice of Numerical Classification*, W.H. Freeman, San Francisco, 1973.
5. H. Späth, *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*, Halsted Press, New York, 1980.



A Use of hierarchical clustering to derive scale-dependent visualizations from a base map.

Scale-dependence is another factor driving the choice of a specific clustering approach. It should be possible to control the amount of graphic detail presented to users so that less detail implies a higher level of abstraction. The fewer clusters used to partition the complete information space, the smaller we can make the visualization's scale (see the sidebar "Hierarchies in Scale-Dependent Visualization").



5 Hierarchical clustering tree for 4,800 SOM nodes. Also shown are three horizontal cuts corresponding to a 10-cluster solution (blue), a 100-cluster solution (red), and an 800-cluster solution (green).

In an interactive setting, users will first be presented with a highly generalized version. Zooming in will reveal more detailed information. One of the challenges in this process is to not simply hide detail, but imbue each generalized version with scale-dependent meaning. Each step of geometric merging must thus be accompanied by the derivation of label terms appropriate to the respective scale level.

A complete hierarchical clustering solution based on the computation of interdocument similarities would suffer from the previously discussed wide range in the number of keywords per document. Instead, I computed a complete clustering solution for the SOM's 4,800 neurons (Figure 5). I then based the membership of individual documents in neuron clusters on the association of documents with neurons (see Figure 3c).

Feature labeling

Labeling individual and grouped features is an integral part of every cartographic depiction. We should give it equal attention when dealing with map-like information visualization.[1] It can be challenging to label a document corpus meaningfully, particularly when the goal of the final visualization isn't to identify documents but to understand the semantic structures and relationships among documents.

Labeling individual documents

As far as the vector-space model is concerned, the set of keywords represents the document. When choosing meaningful document labels automatically, the vector-space model suggests which label terms from the document vectors to select. The keyword that best characterizes a document while distinguishing it from other documents should be the label term. The term weighting formula in Equation 1 expresses this best, since it weighs global term frequency against the frequency of a term within a document:[9]

$$w_{ij} = tf_{ij} \log \frac{N}{df_j} \quad (1)$$

The weight of a term T_j in document D_i is based on the number of occurences of the term within that document (tf_{ij}) and the log of the inverse document frequency

$$(\log \frac{N}{df_j})$$

N is the total number of documents and df_j is the number of documents in which the term appears.

The keyword with the highest weight for a particular document will become its label term. To express the content of each document more meaningfully, I computed three label terms for each document in descending order of term weights.

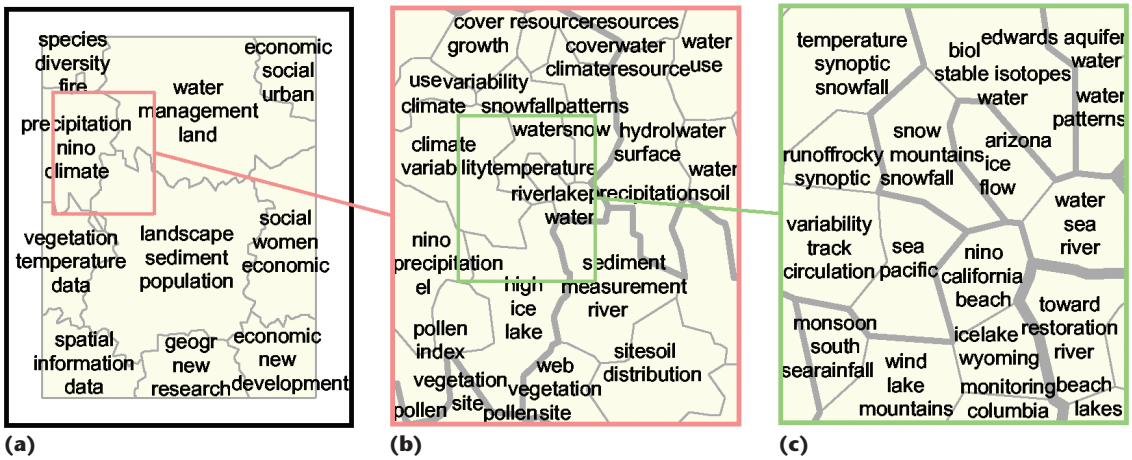
Labeling document clusters

Compared to labeling individual documents, it's more difficult to find a label term that's both representative of a cluster and sets it apart from other clusters. Depending on the chosen algorithm, this may have consequences between two extremes. One possibility would be that the chosen cluster label is specific enough to clearly distinguish clusters from each other. In this case, there's the danger that the chosen term is really only contained in a limited number of documents within the cluster. This raises the question of whether it's then still representative of the cluster. Another possibility is that the cluster label is computed in a manner that ensures that a large proportion of the cluster members actually contain the label term. This raises the likelihood that other clusters might produce the same label. Therefore, the label wouldn't express the cluster's distinct character.

The potential conflict between these two cases becomes more apparent as clusters grow in size. It may in fact be best to deploy two term weighting procedures, one for high-level clusters and another for low-level clusters. For high-level clusters, I found it sufficient to add up term counts within each cluster and choose the term with the highest count to become the cluster label. For low-level clusters, I used a variant of the term weighting formula in Equation 1 that treated each cluster as a superdocument containing all the terms of its member documents.

Visualization

The geographic merging of base-map polygons on the basis of hierarchical clustering and the computation of meaning-bearing labels for the resulting clusters leaves us with a data set to which we can apply traditional principles of cartographic design. For example, when creating a map of counties, we should also display higher level landmark boundaries such as state boundaries. In such strictly hierarchical systems, a higher level boundary always also coincides with a lower level boundary, which means that the added layer of information provides important context without adding much visual



6 Three different zoom levels in a visualization of conference abstracts: (a) complete map shown in a 10-cluster solution and map portions for (b) a 100-cluster and (c) 800-cluster solution. Higher level boundaries are accentuated to provide context during zoom operations.

complexity. When we apply this principle to nongeographic data via hierarchical clustering, it provides for consistency and context during geographic information system (GIS)-like zoom operations (Figure 6). It also makes the simultaneous display of multiple abstraction levels possible, since visual hierarchies can convey cluster hierarchies (Figure 7).

Individual scale-dependent levels

Figure 6 illustrates the relationship between the amount of the displayed total base map area and the visualized level of the hierarchical cluster solution. From left to right, users will first see an overview of the information space consisting of a 10-cluster solution (Figure 6a). Upon zooming in, they see a 100-cluster solution (Figure 6b), and finally an 800-cluster solution (Figure 6c). The three highest ranked labels accompany every cluster.

Multilevel overlay

Compared to other clustering techniques, the nested hierarchy produced by hierarchical clustering is especially advantageous when it comes to displaying a number of different cluster levels simultaneously. However, we must choose map symbols carefully, so that the hierarchical structure is visually conveyed. Figure 7 shows a simultaneous visualization of three levels of the clustering tree, with 10, 25, and 100 clusters, respectively. Cluster labels are ranked according to their computed weight and scaled so that users are first drawn to the higher ranked labels. The point locations of all abstracts are added as faint symbols in the background.

Discussion of results

The resulting visualizations (Figures 6 and 7) allow us to ascertain some advantages and pitfalls of the approach described in this article. In the absence of formal user testing, the following discussion focuses on issues related to the creation and manipulation of the document index and on an interpretation of domain-specific structures and relationships from the visualizations.

Technique

Preprocessing of source data is one of the critical elements in providing successful information visualizations. In this experiment I decided to feed raw term counts to the SOM training procedure. This makes for fairly direct interpretability of component values in the trained SOM and allows the mapping of raw observations onto the SOM. I've experimented with traditional term weighting schemes for the document index, such as the one used for document labeling, but haven't found the results to warrant the added complexity.

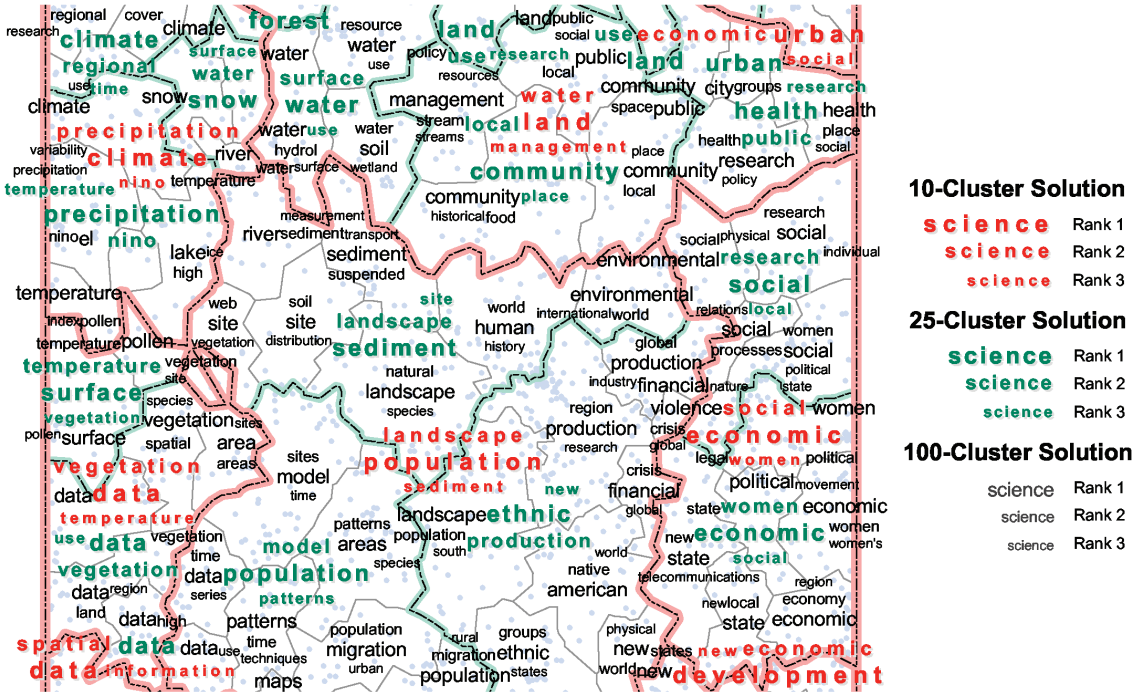
Document length normalization deserves to be revisited. The filtering of the training data set around the mean keyword count led to improved results, compared to previous experiments. Nevertheless, many of the documents with a low keyword count still congregated rather than being distributed across the map. At the clustering stage, this congregation tends to merge too easily with neighboring regions, which leads to an incongruous cluster in the center of the map. Visually, this cluster grows as we move higher in the clustering tree, as indicated by the labels landscape/population/sediment in the 10-cluster solution (Figure 7).

Further stemming would lead to a tighter word index, but we need to balance it with the desire to provide rich cluster labeling. The appearance of the term "new" as a cluster label (Figures 6a and 7) is originally caused by the inclusion of such author-chosen keywords as "New Zealand" or "new world order." After breaking these up into individual components, all abstracts using the term "new" are indexed accordingly. Terms like this should be added to the stop word list, which excludes certain high-frequency terms from the index.

Interpretation

The work presented here is part of a continuing effort to explore the development and state of geographic science. One typical division of the discipline considers the existence of three distinct areas of geographic work:

- Human geography: the study of the human environment, within which urban, transportation, population, economic, and feminist geography are only some of a number of more specific areas of interest and approaches.
- Physical geography: the geographic study of the natural environment, including aspects of geomorphology, climate, vegetation, and so forth.



7 **Visualization of 2,220 conference abstracts with simultaneous overlay of three levels of a hierarchical clustering tree of SOM neurons: 10-cluster solution (red); 25-cluster solution (green); and 100-cluster solution (black). Cluster labels are scaled according to rank within the respective cluster.**

- Techniques: a term that has traditionally referred to work in GIS, cartography, and remote sensing. Rather than merely supporting the work of human and physical geographers, geographic efforts in this area are increasingly part of an emerging cross-disciplinary research field known as geographic information science (GIScience).

Abstract authors tend to use the various subcategories (such as "climate") to identify their particular research topic. The three-tier division emerges implicitly, as related areas of work are arranged in relative proximity. As a result, human geography occupies the right half of the visualization (Figures 6a and 7). Aspects of physical geography dominate the upper left quadrant. The processing and modeling of geographic data dominate the lower left quadrant. The cluster labeled geogr/new/research (Figure 6a) contains many abstracts that deal with the teaching of geography, such as research into the development of new teaching tools and techniques. Issues surrounding resource management dominate the top of the map. This is a heterogeneous area at the intersection of human and natural environments, ranging from the more urban–suburban policy and community questions (note how "land use" appears as "land" and "use" in Figure 7) on the right of Figure 7 to the management of forest and water resources on the left.

Note that geographic concepts and topics dominate the higher level clusters. The names of specific geographic features (such as "Wyoming") appear only as cluster labels at lower levels of aggregation (see Figure 6c).

Combining similarity-based mapping and clustering encompasses the respective strengths and mitigates some of the problems associated with the two approaches. Similarity-based geometric configurations lack the explicit categories necessary for effective communication. While statistical clustering can provide those categories, it lacks the overarching context, particularly with respect to intercluster relationships, that can be provided with a map-like representation. For example, note how areas of geographic research dealing with water tend to be drawn toward each other, even across cluster boundaries. That's why a cluster labeled transport/sediment/suspended (center of Figure 7) is near a cluster labeled water/soil/wetland, across a boundary of two clusters already separated at the 10-cluster level. Population geography is positioned closer to the spatial/data/information cluster (bottom left in Figure 7) because it's the part of human geography in which quantitative modeling and the use of spatially referenced data, like those provided by the US Census Bureau, are most prevalent.

Conclusions

The procedures presented here could be integrated into a fully automated system for visually exploring conference abstracts. One of the remaining issues concerns the mechanism for matching the level of semantic and geometric abstraction to the display scale. (I subjectively chose the display scales for the figures for this article.) I'm currently working on using cartographic generalization principles to determine proper display scales automatically, depending on abstraction level and graphic density.

Thus far, I haven't conducted formal subject testing to investigate how well these visualization techniques

actually work. For this article I chose two extremes: a sparse visualization with a focus on interactivity (Figure 6) and a much richer form, with the potential for static output, leisurely study, and instigation for discussion (Figure 7). In the end, actual solutions should probably fall somewhere in between, providing users with a rich and interactive, yet not overwhelming map.

One of the more persistent comments I've received so far, particularly with respect to the multilevel overlay (Figure 7), is how reminiscent the results are of cartographic depictions of geographic space. The rich labeling and intricacy of cluster outlines make it hard for some to believe that they're not actually looking at geographic structures located on the earth's surface. This is exactly the kind of reaction I'm hoping for, because it raises the possibility that users of a visualization system would be forced to use the same spatio-cognitive skills they employ when dealing with geographic maps. ■

References

1. A. Skupin, "From Metaphor to Method: Cartographic Perspectives on Information Visualization," *Proc. IEEE Symp. Information Visualization 2000* (InfoVis 2000), IEEE CS Press, Los Alamitos, Calif., 2000, pp. 91-97.
2. J. Foley, "Getting There: The Ten Top Problems Left," *IEEE Computer Graphics and Applications*, vol. 20, no. 1, Jan./Feb. 2000, pp. 66-68.
3. H. Couclelis, "Worlds of Information: The Geographic Metaphor in the Visualization of Complex Information," *Cartography and Geographic Information Systems*, vol. 25, no. 4, Oct. 1998, pp. 209-220.
4. S. Fabrikant and B. Buttenfield, "Formalizing Semantic Spaces for Information Access," *Annals of the Assoc. American Geographers*, vol. 91, no. 2, June 2001, pp. 263-280.
5. J. Wise et al., "Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Documents," *Proc. IEEE Symp. Information Visualization 1995* (InfoVis 95), IEEE CS Press, Los Alamitos, Calif., 1995, pp. 51-58.
6. H. Chen, C. Schuffels, and R. Orwig, "Internet Categorization and Search: A Self-Organizing Approach," *J. Visual Comm. and Image Representation*, vol. 7, no. 1, Mar. 1996, pp. 88-102.
7. D. Rushall and M. Illgen, "DEPICT: Documents Evaluated as Pictures," *Proc. IEEE Symp. Information Visualization 1996* (InfoVis 96), IEEE CS Press, Los Alamitos, Calif., 1996, pp. 100-107.
8. T. Kohonen et al., "Self Organization of a Massive Text Document Collection," *Kohonen Maps*, E. Oja and S. Kaski, eds., Elsevier Science, Amsterdam, 1999, pp. 171-182.
9. G. Salton, *Automated Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*, Addison-Wesley, Reading, Mass., 1989.
10. X. Lin, "Visualization for the Document Space," *Proc. IEEE Visualization 92*, IEEE CS Press, Los Alamitos, Calif., 1992, pp. 274-281.

André Skupin *is an assistant professor of geography at the University of New Orleans. His research interests include text document visualization, geographic visualization, cartographic animation and hypermedia, and cartographic generalization. He received a Dipl.-Ing. degree in cartography from the Technical University Dresden, Germany (1992) and a PhD in geography from the State University of New York at Buffalo (1998). He did graduate research with the National Center for Geographic Information and Analysis (NCGIA) and has worked in the GIS industry in the US, Germany, and South Africa.*

Readers may contact André Skupin at the Dept. of Geography, 261 Liberal Arts Bldg., Univ. of New Orleans, New Orleans, LA 70148, email askupin@uno.edu.