Incremental Computation of Information Landscapes for Dynamic Web Interfaces

Vedran Sabol,[1] Kamran Ali Ahmad Syed,[2] Arno Scharl,[3] Markus Muhr[1] and Alexander Hubmann-Haidvogel[3]

[1] Know-Center Graz, Division for Knowledge Relationship Discovery
Inffeldgasse 21a, 8010 Graz, Austria
{vsabol, mmuhr}@know-center.at

[2] Vienna University of Economics and Business, Institute for Information Business,
Augasse 2-6, 1090 Vienna, Austria
syed.k.a.a@gmail.com

[3] MODUL University Vienna, Department of New Media Technology
Am Kahlenberg 1, 1190 Vienna, Austria
{arno.scharl, alexander.hubmann}@modul.ac.at

ABSTRACT

This paper presents a technique for the visual analysis of topical shifts in dynamically changing textual archives. Our approach is based on the well-known information landscape metaphor, whereby topical changes are represented by changes in landscape topography. Incremental clustering and multi-dimensional scaling algorithms are periodically applied to a changing document set for generating a series of information landscapes. The resulting landscapes are suitable for dynamic Web interfaces, enabling the user to explore topical relationships and understand topical shifts and trends in changing document repositories.

Keywords

Information landscapes, information visualization, visual analytics, document clustering, multi-dimensional scaling.

INTRODUCTION

Topical analysis of large text document repositories is a complex task which can be addressed by quantitative methods such as document clustering, by visual methods such as information landscapes, or a combination thereof. The fact that large text repositories from Web-based sources are often dynamic in nature further complicates this challenging task. As new documents are added and old documents are removed, topical clusters and topical relationships in the repository change accordingly.

In this paper, we present a prototype for visualizing topical changes using information landscapes with dynamic topographies. Information landscapes convey topical relatedness of visualized documents through spatial proximity in the visualization. We extend this approach by representing temporal changes in the document repository through modifications of the topography of corresponding regions in the information landscape.

The rest of the paper is organized as follows: After a brief discussion of related work we introduce the information landscape with dynamic topographies and discuss their application in a use case involving an environmental document set. The subsequent section presents the algorithms, which are applied on a changing document repository to incrementally compute a series of information landscapes. We conclude by discussing the advantages and disadvantages of our approach and provide an outlook for incorporating our method into an existing visual Web interface.

IHC 2010 – IX Simpósio sobre Fatores Humanos em Sistemas Computacionais. October 5-8, 2010, Belo Horizonte, MG, Brazil.

RELATED WORK

Information landscapes are commonly used to visualize topical relatedness in large document repositories, for example in Krishnan et al. [9] and Andrews et al. [2]. Static landscape visualizations, however, cannot convey changes. ThemeRiver [7] is a visual representation designed to represent changes in topical clusters, but it cannot express relatedness between documents or topical clusters. Visualization of topical changes through information landscapes with dynamic topologies were proposed in Sabol, Granitzer and Kienreich [13], albeit only for small data sets, and later extended in Sabol and Scharl [14]. An approach suitable for larger data sets was introduced in Sabol et al. [15] and demonstrated in Sabol and Kienreich [16]. It relies on 3D acceleration for animated morphing of landscape geometry, which makes it unsuitable for Web applications. A survey of visualizations for time-oriented visualizations can be found in Aigner et al. [1].

DYNAMIC INFORMATION LANDSCAPES

Geographic Map Metaphor

Information landscapes are visual representations based on a geographic map metaphor. They are used for analyzing relationships in large, high-dimensional data sets by con-

veying relatedness in the data through spatial proximity in the visualization. We apply the concept of information landscapes to visualizing topical relationships in text document sets. Hills and islands represent groups of topically related documents, which emerge where the document density is large. They are separated by sparsely populated areas which are represented as oceans or valleys. To provide orientation and meaning, peaks are labeled with terms describing the corresponding topical clusters. The height of a hill serves as an indicator for the amount of documents belonging to the topic, while its compactness is an indicator of topical cohesion.

Dynamic Topography

When computed incrementally, dynamic information landscapes represent topical changes in the underlying document repository as tectonic processes that modify the landscapes' topography. Growing or shrinking hills and islands indicate emerging or fading topics, respectively. Their moving towards or apart from each other indicates the convergence or divergence of topical clusters. Changes in the data set are incorporated in the landscape incrementally so that regions which are not affected by the data set changes preserve their position and shape. This allows the user to immediately identify regions experiencing changes, and put them into context through the recognition and orientation provided by the already known, unchanged elements of the topography.

An example of an incrementally generated landscape with dynamic topography is shown in Figure 1. The first in the resulting series of landscapes was constructed based on 20,000 articles from a climate change document archive. Each week, about 1000 to 2000 documents (5-10%) from Web sources were appended to this archive (approximately the same number of documents was removed), to compute the other three landscapes. As seen in Figure 1, mountains around region B remain almost unchanged over time showing an approximately constant number of documents belonging to the corresponding topics. Slight label fluctuations can nevertheless be observed. Mountains south-east of regions C and D are slowly gaining dominance, in particularly an increase is visible between the second and third landscape. However, these trends do not continue as both topical clusters shrink again slightly in landscape 4.

If the majority of documents belonging to a topical cluster are removed then the associated mountain, for example in the north-east of the region D in the first landscape, is deformed and may even disappear. On the other hand, if the effect of adding new documents to a topical group is dominating then the mountain regions become more established solidifying and protruding, for example between landscapes 2 and 3 in the south-east of region D. The net effect of adding and removing of documents may also result in complete topographical reconfiguration of a region, for example around A, indicating a major change in the repository. As opposed to that small movements of mountains towards or apart from each other are signs of topical convergence or divergence of clusters.

METHOD

The incremental algorithm for computing dynamic information landscapes operates on high-dimensional document term vectors. These are generated by co-occurrence analysis [10] and significant phrase detection [6] algorithms. Cosine coefficient is used to compute the similarity between term vectors in all stages of the algorithm. Our algorithm consists of the following five consecutive steps:

1) *Document Clustering*. The document set is clustered into groups of topically related documents using an incremental k-means algorithm.
2) *Cluster Positioning*. Topical clusters are projected into the 2D visualization space according to their topical similarity using a force-directed placement algorithm.
3) *Document Positioning*. Documents are projected into the 2D space depending on their topical similarity to the clusters using a fast interpolation technique.
4) *Map Rendering*. Landscape image is generated by computing an elevation matrix and assigning colors to pixels depending on local elevation values.
5) *Label Computation*. Labels describing major landscape peaks are high-frequency keywords belonging to documents placed in the neighborhood of these peaks.

Document Clustering

Spherical k-means algorithm [5] is used to partition the document set into topical clusters. As k-means is known to be sensitive to the initial centroid configuration we use the k-means++ seeding method [3]. A strategy for splitting and

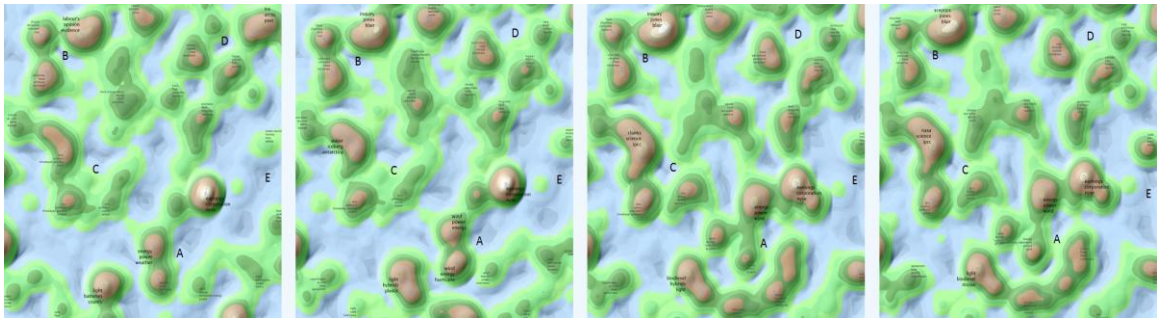

Figure 1: Initial landscape based on 20,000 documents (left) and incremental versions after one, two and three weeks

merging of clusters [11] is used for guessing the number of clusters within the specified minimum and maximum bounds. Due to usability considerations, which include human cognition limits and the perception of what is an aesthetically pleasing information landscape, we typically set these limits to 20 and 40, respectively. After running k-means, splitting of large clusters with low cohesion and merging of small, similar clusters will be performed, as long as this brings an improvement according to the Bayesian Information Criterion [12]. After splitting and merging several k-means iterations will be performed to refine the partition. The algorithm scales with O(*rn*), where *r* is the number of clusters and *n* is the number of documents. As there is an upper limit to r the algorithm can be considered to scale linearly with n.

In the incremental case, an existing partition of the document set is used as the initial state. Deleted documents are removed from their clusters; new documents are added to the most similar cluster centroid. k-means, including split and merge procedure, is then applied to refine the partition.

Cluster Positioning

To project the high-dimensional cluster centroids into the 2D visualization space we use a force-directed placement (FDP) algorithm [2]. Attractive forces pull together topically similar centroids, while dissimilar centroids are pushed apart by repulsive forces. As a consequence, the iterative FDP algorithm will asymptotically converge towards a stable layout where spatial closeness between centroids is related to their topical relatedness. The advantage of FDP is that it produces good and aesthetically pleasing layouts. The disadvantage is that it scales poorly – the variant we use has a $O(r^3)$ running time. However, due to the fact that the number of clusters has an upper limit, the running time for cluster positioning can be considered constant.

An important FDP feature is that it is inherently incremental when applied on a previously computed stable layout. Incremental clustering impacts similarities between cluster centroids. Re-applying FDP will modify the previous layout to reflect the modified similarities.

Document Positioning

We introduce a fast document positioning procedure where document 2D positions are computed based on centroid 2D positions only. First a Delaunay triangulation [4] of the clusters $\{c_1, c_2, \ldots, c_r\}$, obtained in the cluster positioning step, is performed. Note that for simplicity we use *c* to represent both a cluster and its position in 2D. For each Delaunay triangle (c_i, c_j, c_k), we partition the cluster set into three disjoint lists: each triangle vertex is assigned a list of all clusters (including itself), which are more similar to that vertex (cluster) than to the other two vertices (clusters) of the triangle. Given a document *d* belonging to the cluster c_k, a triangle is chosen that maximizes the sum of the similarities between *d* and the most similar cluster in each of the three lists. The triangle choice therefore considers all clusters rather than only the immediate neighborhood. The document is positioned within the chosen triangle using Barycentric coordinates [17], where masses assigned to the triangle vertices correspond to the similarities between the document and the most similar cluster from the vertex list.

Delaunay triangulation of *r* centroid vertices runs in $O(r*log(r))$, and the construction of the triangle data structures (cluster lists) in $O(r^2)$ time. After this preprocessing, searching for a triangle and computation for maximum similarity values in the triangle's cluster lists for a given documents is in $O(r)$ time. However, due to the upper limit on *r*, document positioning can be considered to scale linearly with the number of documents.

Map Rendering

To compute the landscape image an elevation matrix is computed, with dimensions that correspond to the image resolution of 4096 x 4096 pixels. Each document can be thought of as a small peak, so that in areas where document density is large the peaks are superimposed adding to the elevation values of the underlying matrix cells. Image pixels are assigned colors depending on the density of the corresponding density matrix cells. We use blue to express lowest density, then green and brown, and finally light gray for highest density. The resulting image resembles a geographic map with peaks at areas where document density is large, while low density areas will be represented as oceans or valleys. For a fixed resolution of the image, the algorithm scales linearly with the number of documents.

Label Computation

A kernel window base peak detection algorithm applied on landscape elevation data is used for spotting mountains with significant peak heights. Documents associated with each mountain are identified using minimum Euclidean distance criterion. Terms with highest occurrences in the associated document vectors are chosen as labels for the mountain. Finally, labels are drawn on the landscape image at positions of their corresponding mountain peaks. For an elevation matrix of a fixed size and an existing upper bound on number of clusters, label computation scales linearly with the number of documents.

Incremental Computation

To compute an initial landscape, the algorithm is applied on the whole data set. When the data set changes the algorithm is reapplied using a previously computed stable partition and layout to initialize the clustering and FDP algorithms. This is necessary because both algorithms are very sensitive to initial conditions, and otherwise they would most likely yield completely different result. Nevertheless, to guarantee that the incremental landscape changes are smooth the algorithm must be reapplied when the data set change reaches a threshold (typically up to ten percent).

As all steps of the algorithm scale no worse than linearly with the number of documents, time complexity of the whole process can be considered $O(n)$. Table 1 shows running times for the initial non-incremental run and for the

following three incremental steps (note the reduced running times for clustering and FDP in the incremental case).

	Initial 20000	+/- 1000	+/- 1000	+/- 1000
1. Clustering	1495	1065	1035	1035
2. Cluster pos.	108	21	22	21
3. Document pos.	16	15	15	16
4. Map rendering	27	25	26	25
5. Labeling	5	5	5	5
Total time (sec.)	**1651**	**1131**	**1103**	**1102**

Table 1: Running times for the initial 20,000 documents and three incremental steps, each with 5% added and removed documents (3GHz Core2 Duo, JVM 1.6.0_20).

CONCLUSION AND FUTURE WORK

The incrementally computed landscapes will be included as a visual navigational aid into the *Media Watch on Climate Change* [8], an environmental news and blog aggregator publicly available at http://www.ecoresearch.net/climate. The portal currently includes a static landscape displayed via an OpenLayer client, using a JavaScript framework to synchronize the landscape with other semantic and geographic visualizations. In this multiple coordinated view setup, the user will be able to navigate back and forth in time through time interval sliders. Including a temporal dimension into the Web-based navigation via dynamic information landscapes will help understand the structure and topical shifts of the underlying knowledge repository.

ACKNOWLEDGMENTS

Algorithms and visualization methods presented in this paper were developed within the RAVEN project (www.modul.ac.at/nmt/raven), which is funded by the Austrian Research Promotion Agency within the strategic objective FIT-IT Semantic Systems.

REFERENCES

1. Aigner, W., Miksch, S., Müller, W., Schumann, H., Tominski, C., Visualizing Time-Oriented Data – A Systematic View, *Computers and Graphics*, 31(3), pp. 401-409, 2007.
2. 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), pp. 166–181, 2002.
3. Arthur, D., Vassilvitskii, S., k-means++: the advantages of careful seeding, *in SODA*, N. Bansal, K. Pruhs, and C. Stein, Eds. SIAM, pp. 1027–1035, 2007.
4. de Berg, M,. Cheong, O., van Kreveld, M., Overmars, M., *Computational Geometry: Algorithms and Applications*. Springer-Verlag. (2008).
5. Dhillon, I.S., Modha, D.S., Concept decompositions for large sparse text data using clustering, *Machine Learning*, vol. 42, no. 1/2, pp. 143–175, 2001.
6. Hart, M., Bautin, M., Significant Phrases Detection, *State University of New York*, Tech Report, 2007.
7. Havre, S., Hetzler, E., Whitney, P. and Nowell, L., ThemeRiver: Visualizing Thematic Changes in Large Document Collections", *IEEE Transactions on Visualization & Computer Graphics*, 8(1), pp. 9-20, 2002.
8. Hubmann-Haidvogel, A., Scharl, A. and Weichselbraun, A. (2009). "Multiple Coordinated Views for Searching and Navigating Web Content Repositories", Information Sciences, 179(12): 1813-1821.
9. Krishnan, M., Bohn, S., Cowley, W., Crow, V., Nieplocha, J., Scalable Visual Analytics of Massive Textual Datasets, *21st IEEE Int'l Parallel and Distributed Processing Symposium*. IEEE Computer Society. 2007.
10. Liu, W., Weichselbraun, A., Scharl, A., Chang, E., Semi-Automatic Ontology Extension Using Spreading Activation. *Journal of Universal Knowledge Management*, vol, 0, no. 1, pp. 50-58, 2005.
11. Muhr, M., Granitzer, M., Automatic Cluster Number Selection using a Split and Merge K-Means Approach, *in Proceedings of the 2009 20th International Workshop on Database and Expert Systems Application*, pp. 363-367, 2009.
12. Pelleg, D., Moore, A., X-means: Extending K-means with efficient estimation of the number of clusters, *in Proceedings of the 17th International Conf. on Machine Learning*, pp. 727–734, 2000.
13. Sabol, V., Granitzer, M., Kienreich, W., Fused Exploration of Temporal Developments and Topical Relationships in Heterogeneous Data Sets, *Proceedings of the 11th International Conference Information Visualization*, pp. 369-375, 2007.
14. Sabol, V. and Scharl, A. "Visualizing Temporal-Semantic Relations in Dynamic Information Landscapes", *11th International Conference on Geographic Information Science*, Semantic Web Meets Geospatial Applications Workshop. Girona, Spain: AGILE. (2008).
15. Sabol, V., Kienreich, W., Muhr, M., Klieber, W., Granitzer, M., Visual Knowledge Discovery in Dynamic Enterprise Text Repositories, *in Proceedings of the 13th International Conference on Information Visualisation*, pp. 361-368, 2009.
16. Sabol, V., Kienreich, W., Visualizing Temporal Changes in Information Landscapes, *Poster and Demo at the EuroVis*, 2009.
17. Weisstein, E. W. "Barycentric Coordinates" from MathWorld - A Wolfram Web Resource. mathworld.wolfram.com/BarycentricCoordinates.htm