Concept Map:

Visualizing Data Clusters in Categorical Domains

David Rouff
Department of Computer Science, University of Maryland
College Park, MD 20770
dcrouff@umd.edu

ABSTRACT

No visualization currently enables both a clear overview and details of associations between elements in large categorical data sets. Many approaches have been taken to address the problem, but each has produced limited results due to the inherent challenge in visualizing many-dimensional data sets. In computer and social networks, for example, every node has potential connections to each of the other nodes, and so every node is a dimension in the dataset. Many visualizations, including the popular node-link representation, quickly become occluded as networks grow beyond a trivial size.

While no single visualization has succeeded in providing a complete insight into categorical data, several approaches are successfully used to view specific aspects of the domain. Interesting features that are occluded in one visualization may become readily apparent in another. With this understanding, the Information Visualization tool developed in conjunction with this research provides multiple coordinated views that present an organized visualization into categorical data sets. Each view takes a different approach to identifying clusters in the domain, and each uses size, position, color, and shading elements to communicate associations within and between clusters. Combined with the interactive features to explore and dynamically alter the canvas, the visualization system is named ConceptMap. The application includes two main innovations: First, a modified approach to hierarchical clustering that arranges nodes within a cluster at the same time it aggregates higher levels of cluster groupings. And second, A dendrogram coordinated with a reordered adjacency matrix in a fashion that the overview, clusters, and elements are visible simultaneously.

Keywords:
Information Visualization, Dendrogram, Hierarchical Clustering, Adjacency Matrix, Categorical Data, Tabular Data

1.INTRODUCTION/MOTIVATION

Large, high dimensional data sets are inherently difficult to understand and categorize into groupings that can be expressed and understood visually. Information visualization techniques allow an increasingly larger volume of dimensions to be communicated by using increasingly sophisticated combinations of size, color, location, shape, texture, and other basic visual elements. But the dimensionality communicated is still small compared to the high dimensionality of contemporary data sets such as social networks or text corpuses.

The tool presented here offers a combination of data visualizations that abstract the dimensionality away by clustering objects and producing a similarity based ordering. While data element dimensions are not directly represented in the visualizations, the increasing number of matching dimensions between objects causes the objects to cluster increasingly closer together. Through such clustering, a complex data set can be analyzed and interesting features identified.

The tool, called ConceptMap, was created to both display a novel combination of information visualization techniques, and to explore the corpus of published research in the information visualization field. The primary data set analyzed, therefore, is the set of terms listed as keywords from information visualization related papers published by the ACM.

Following the introduction, the paper is organized as follows: Section 2 recaps related research into both clustering and visualizations. Section 3 describes the functional design and novelties in ConceptMap. Section 4 describes the technical details of the clustering and visualization algorithms. Section 5 analyzes the visualizations created by the tool, to highlight interesting results in the clustering of the Information Visualization corpus of research. Section 6, in order to show the ability to apply the ConceptMap design to other data sets, analyzes

visualizations of Senate voting records. Finally, section 7 proposes future research and refinements to the application.

2. RELATED WORK

The content of published research is inherently a categorical domain. The relationships among key concepts is not easily discovered or represented in a visualization. Many approaches have been taken to the problem of showing the relationships among categorical data sets, a task isomorphic to visualizing computer and social networks. Several challenges exist in the network visualization area. One of the chief problems is the many-dimensional nature of the data. Every new node in a network has potential connections to each of the other nodes, and so every node adds a dimension to the dataset. In many networks, there is no inherent order to the nodes, which poses a problem in arranging the data following an order that viewers will recognize. The lack of order, however, also grants the visualization tool an amount of freedom to rearrange the data in order to make it more understandable.

Research into this area includes several types of visualizations including self-organizing maps, node-link diagrams, and adjacency matrices. This is an open field of research because the existing visualizations are able to communicate an aspect of the categorical data, but none have yet been able to communicate all aspects of the data. Of the published research, the following papers related the most useful and novel approaches:

In, "A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations," by Mohammad Ghoniem, Jean-Daniel Fekete, and Philippe Castagliola, the concept of using an adjacency matrix is presented and compared with a node-link diagram.[1] While the node-link diagram is more understandable by novice viewers, it becomes positively unreadable as the number of nodes and links gets larger than a trivial set. Their adjacency matrix is useful for looking row by row (or column by column) at nodes to see their connectivity. This paper did not address re-ordering the adjacency matrix, so their adjacency matrix is not very useful for looking across the entire network by itself.

Harri Siirtola published several important works in this area, and was the first attempt to organize an adjacency matrix by permuting the rows and columns. Applications of Siirtola's reorderable matrix are discussed across several papers, including his own work, "Interaction with the Reorderable Matrix," [2] and "The Barycenter Heuristic and the Reorderable Matrix," co-authored by Harri Siirtola and Erkki Mäkinen. [3]

In more recent research, the MatrixExplorer application offers coordinated views between a clustered adjacency matrix and a node-link representation to depict social networks [4]. As Nathalie Henry and Jean-Daniel Fekete describe, the crosswalk between visualizations offered in MatrixExplorer provided key insights to the sociologist users of the tool. The success in presenting multiple visualizations reinforced the similar approach developed into ConceptMap.

The multiple visualizations of this approach, as well as the hierarchical clustering method to approximate the optimal ordering of elements in less than non-deterministic polynomial time came from the Hierarchical Clustering Explorer application by Jinwook Seo and Ben Shneiderman [5]. An earlier work, by Stephen P. Borgatti [6] was also instrumental for the hierarchical clustering algorithm. For the adjacency matrix and dendrogram, this work adopts the hierarchical clustering approach, but a new distance metric was designed to measure the similarity of relationships between categorical data elements.

Papers on keyword extraction and vectorization of documents were also explored. Of note, Seung-Shik Kang [7] provided a novel method of weighting the keywords based on both the document frequency and the term frequency. This allows better vectorization of documents and could lead to an improved representation. In [8], Lance Parsons describes several approaches to cluster high dimensional data. This document was also referenced for dimensionality reduction using feature transformation and feature selection.

A unique representation that could be directly applied to the dendrogram was presented in "HD-Eye: Visual Mining of High Dimensional Data"[9] The HD-Eye visualization is a 3D surface where the height of the graph is proportional to the density of the cluster. This novel representation allows users to quickly estimate the cardinality of the cluster they are choosing.

3. APPROACH TO THE CONCEPT MAP DESIGN

Relationships that are occluded in one visualization may be readily apparent in another. With a goal of visualizing complex categorical data, the core design feature of ConceptMap is providing multiple visualizations. The two interdependent visualizations take highlight clusters of relationships in distinct ways. Each uses visual elements of size, position, color, and shading elements to communicate associations within and between clusters. Also, each visualization provides interactive features to explore and dynamically alter the graphic.

The sample domain used in the application is the collection of keyword concepts listed in ACM published papers. Research concepts are words, and many visualizations such as TextArc reveal how difficult it is to make useful visualizations that contain many words [10]. Likewise, many visualizations such as Vister, [11] show the limited value of node-link visualizations when the number of nodes and links grows beyond a relatively trivial size. Apart from

the limited success of node link applications, several studies show the node link limitations in the abstract [1]. So, the approach in ConceptMap is to aggregate data up to "meaningfully sized" clusters based on the size of the canvas and relative volume of published work about the concept. "Meaningfully sized" is relative to the user's needs, and is therefore user configurable so that a completely detailed map will be available by setting the aggregate level to 1.

The atomistic level is the list of concepts, which is taken as the list of keywords from the metadata of information visualization research papers available from the Association of Computing Machinery (ACM). Taken together, the keyword lists of published research represent a categorical data set. This data shows variations in the frequency of various concepts, with many terms repeated between papers that discuss different aspects of the same topics.

- This data provides a means to define variations in the popularity of terms, by the number of times each term is listed as a keyword across the set of papers.

- Associations between different terms in the set are defined by the concepts appearing together within a piece of published research. This conclusion forces the assumption that there is a unity in theme of each published paper, and therefore that the key terms within a paper are related.

As the data is processed, related concepts cluster together to build a map of associations and distinctions. The user interface of the design includes two frames, the larger frame being a canvas of the visualization and the smaller frame containing a series of menu items and filters.

The design also includes interactive features, including an interactive visualization that allows users to click on clusters to see their contents in the menu frame.

4. CONCEPT MAP IMPLEMENTATION

The Dendrogram and Clustered Adjacency Matrix visualizations of ConceptMap were coded in Matlab. The core functionality in creating both the clustered adjacency matrix and the Dendrogram is a hierarchical clustering algorithm tailored with a distance metric designed to calculate distance as the similarity of values between rows or columns of an adjacency matrix.

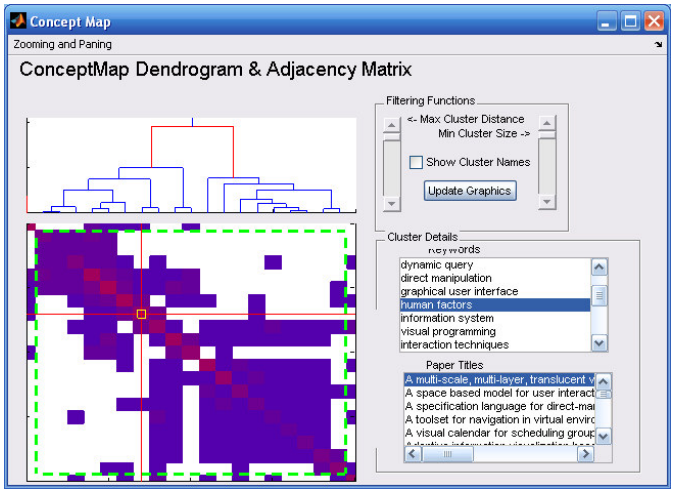


Figure 1. User Interface of the Dendrogram and Adjacency Matrix ConceptMap Application

The algorithm first processes the source data of papers and keywords to build an adjacency matrix (A). Matrix A is populated by traversing all keywords listed across all papers. Defining N as the number of unique keywords, the cardinality of A is N by N. For the sake of processing speed and matrix readability, the cardinality has been trimmed down in this application to show only keywords that exist in multiple papers, 338 keywords out of the total set of 1677. For each pair of keywords listed in a paper (arbitrarily, the keyword represented by row/column i and the keyword represented by row/column j), the value at Aij is incremented by 1.

The clustering algorithm uses 2 additional data structures: a distance table, storing how similar/different 2 rows are, and a 'cluster' data structure, containing the data fields necessary to build the hierarchy of clusters in a binary tree.

The clustering algorithm was adapted from the design employed in Hierarchical Clustering Explorer and a general clustering design described by Stephen P. Borgatti in [8]. Initially, each row is made into its own cluster. A distance table is created, storing the distances between each of the clusters.

Hierarchical clustering requires a distance metric to decide which clusters are nearest to each other. Euclidean distance is meaningless in a categorical dataset, so distance is calculated as a measure of similarity between clusters in the adjacency matrix. The similarity is calculated to be a value between zero and one. The smaller the value, the more alike the two rows are in their values (i.e., there's less distance between them). Each location in the distance matrix, distance(x,y), is calculated as the distance from cluster x to cluster y with the following equation:

$$Distance(x, y) = \frac{\sum_{i=1}^{N} |AdjacencyMatrix(x,i) - AdjacencyMatrix(y,i)|}{\sum_{i=1}^{N} |AdjacencyMatrix(x,i) + AdjacencyMatrix(y,i)|}$$

At each pass of a clustering loop, the two clusters with the minimum distance are identified and merged together. In the Adjacency Matrix, the rows that were merged together are deleted, and a new row representing the new merged cluster is added. The value of the new cluster is calculated as a weighted average of the nodes within that cluster, i.e., a cluster of 5 rows merged with a cluster of 2 rows will have a value of (5*Cluster1 + 2*Cluster2)/7. During the clustering process, the distances between merged clusters is saved in the data structure and used to scale the height of each bar in the dendrogram.

The visual quality of performance of the general clustering algorithm was improved quite dramatically by the addition of a step to evaluate the four distinct ways that two clusters can be merged. By comparing the left and right ends of each cluster in a "First, Outer, Inner, Last" (FOIL) fashion, the clusters were merged with the most similar edges touching. This involved additional computing time to calculate the four distances and frequently reverse the order of one of the clusters, but the impact is visible in the following before/after images of the adjacency matrix. The matrix is less "noisy" in the after image, but the noise is migrated significantly closer to the diagonal where the clusters are identified. This effect is highlighted in figure 2.

Storing a separate distance table saves the work of re-calculating all distances after every clustering merge. After each clustering pass, only the distances between the new cluster and the remaining clusters needs to be calculated. The clustering loop continues until only one cluster remains.

After the hierarchical clustering process, the leaf nodes of the cluster data structure are traversed in order to extract the permutation vector. The last node added is inherently the root of the dendrogram binary tree. Starting with this node, the algorithm drills down to the leftmost child first, then walks to the right, going up and down the tree to visit every leaf node.

This permutation vector is used to reorder the rows and columns of the original Adjacency Matrix, causing the most affiliated elements to appear as clusters of higher values along the diagonal.

The visualization displays the dendrogram aligned over the adjacency matrix, so the user can compare values between them.

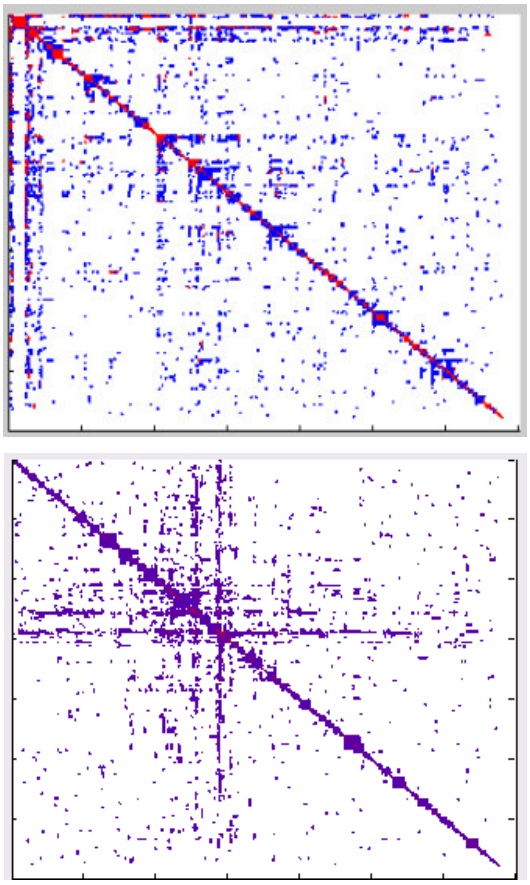


Figure 2. Adjacency Matrix Before (top) and After (bottom) adding the FOIL calculation to clustering

In the visualization, the quantity in each matrix element is represented by colors, with white showing empty space, blue showing 1 association, and colors growing from blue to red as the number of associations increases up to a maximum value of 38. The dendrogram levels are blue, with exception of the root of a cluster, that is signified by being drawn in red.

The application's interface contains controls to affect the graphics displayed and the algorithm's performance. The following list highlights key functionality:

- First, a slider allows users to define the minimum number of closely associated elements to be treated as a cluster instead of outliers between clusters.

- Next, a slider allows users to define the maximum distance between elements within a cluster. This is best visualized as the height of the dendrogram bar containing a cluster.

- When populated with many small clusters, the names obscure the visualization, so a checkbox allows users to show and hide the cluster names.

- Clicking on a cluster filters the list of keywords in the listbox, and clicking on a keyword in the listbox causes the

keyword's location to be highlighted in the matrix and dendrogram.

- A drop-down menu allows users to zoom into the visualizations and pan across them, while they remained linked by keyword.

- Finally, an 'Update Graphics' command button applies the changes from the controls and re-executes the clustering algorithm.

5. ANALYSIS OF THE VISUAL FEATURES IN THE DENDROGRAM AND ADJACENCY MATRIX

The significant volume of white space in the upper and lower triangular matrices just a few steps away from the diagonal shows the disconnected breadth of research in the field. The original corpus of papers included 1677 distinct keywords. Even after trimming the cardinality down to the 338 keywords that exist in multiple papers, the adjacency matrix is extremely sparse. The interesting visualizations, therefore, occur along the diagonal in the matrix. Noise off of the diagonal and three distinct types of clusters are described in the following subsections.

5.1. STRONG CLUSTERS

The adjacency matrix visualizes strong clusters as solid rectangles. The rectangle indicates that every keyword is associated through published papers to every other keyword. A node-link diagram of these terms would be fully populated. A cluster is made stronger if it has a solid border of white (empty) space around it. In addition to every term having a strong relationship with every other term, the presence of white space around the cluster indicates that other concepts are not related to the terms in this cluster. The branching pattern in the associated dendrogram shows that the cluster is built incrementally, in most levels up the dendrogram tree add one keyword to either side of the cluster. In only the root level is a second cluster of size 2 added to the overall cluster.

In the order they appear along the diagonal, the example given in the figure below shows the cluster of the following terms: *'empirical evaluation', 'color', 'scientific visualization', 'icon', 'preattentive', 'target detection', 'cognitive psychology', 'boundary detection', 'human vision', 'orientation', 'multidimensional data',* and *'multivariate data.*'

The overview of the adjacency matrix shows that this data set contains few strong clusters.

5.2. BIMODAL CLUSTERS

Bimodal clusters are identified by a characteristic sideways hourglass shape. In this cluster, one or more keywords are strongly associated with two groups that have nothing else in common. The distance calculation between clusters is sufficiently reduced by the association with keywords in the hourglass middle, causing the unfamiliar clusters to be drawn together. The branching pattern in the associated dendrogram shows that the cluster is built as 2 clusters that are pulled together by the elements in the hourglass middle.

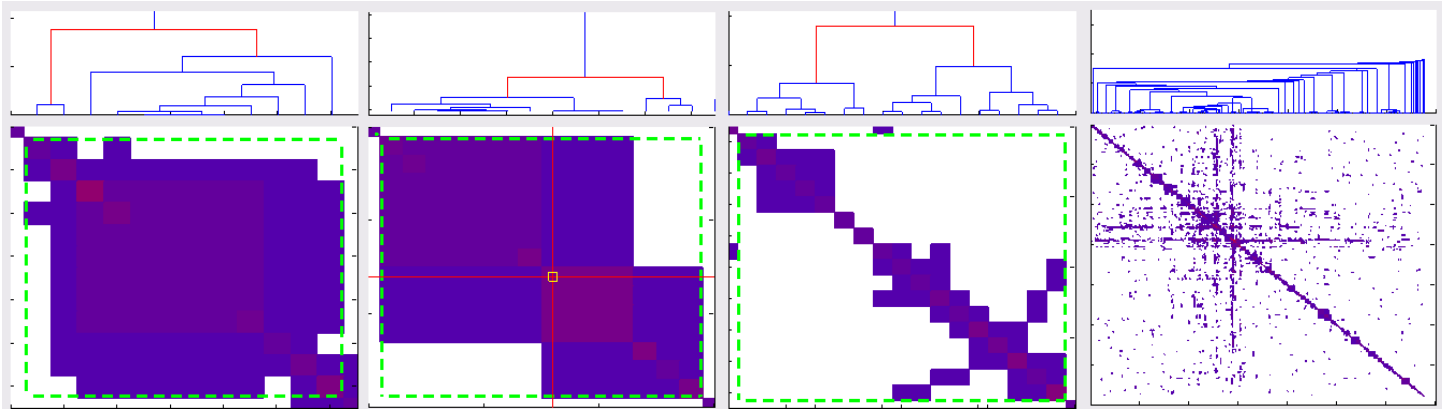
In the example bimodal cluster, the terms: *'information presentation', 'expressiveness', 'presentation tool', 'effectiveness', 'composition algebra', 'automatic generation',* and *'graphic design' are linked to 'empirical study', 'overview',* and *'3D information visualization.'* The hourglass center drawing the sub-clusters together consists of: *'Adobe Acrobat'(*Highlighted with the red + and yellow square*), 'Windows Media Player', 'Real Player',* and *'QuickTime.'*

This hourglass offers the insight that information visualization papers about Acrobat and the other products were either about the cluster of terms on the left side (beginning with *'information presentation'*) or the right side (beginning with *'empirical study'*).

5.3. WEAK CLUSTERS

With the description of bimodal clusters above, weak clusters can be thought of as multi-modal. Each term is associated strongly with the terms immediately to either side, but not to the rest of the terms in the cluster. The dendrogram above the adjacency matrix depicts this more clearly, where the weak cluster is really a set of rather distantly related entities. This adjacency matrix shows the sparseness of relationships between terms not immediately adjacent. In order on the diagonal, the keywords are: *'occlusion'* (no pun intended!)*, 'data compression', 'edge detection', 'image databases', 'searching'. 'self organizing maps', 'research', 'spreadsheet programs', 'complex data', 'information visualization system', 'data types', 'expert systems', 'software engineering', 'case study', 'software visualization', 'color graphics',* and *'object-oriented'* When calculating for clusters across categorical data, this result shows the need to visually analyze the clustering result and tailor the maximum cluster distance and minimum cluster size to identify the largest and most complete rectangles possible in the data set.

The combination of adjacency matrix and dendrogram offers an insight into the nature of clustering. The matrix shows a strong cluster as a solid rectangle and shows how the cluster is built up from a core association by adding one node to the core at a time in a manner analogous to the way a snowball is rolled into a snowman. The weak cluster's



1. Strong Cluster *2. Bimodal/Hourglass Cluster* *3. Weak Cluster* *4. Noise in the Map*

Figure 3. Comparison of Cluster Types.

dendrogram, by contrast, shows a high "branching factor" where two elements or clusters are put together at each level. The balanced nature of the resulting binary tree appears to be a way to indicate that the cluster is weak and the elements do not really belong together.

This insight opens the possibility for identifying clusters through an approach that does not rely on a minimum size and maximum distance. Rather, clusters of varying size can be identified by searching for the snowball effect.

5.4. NOISE

The dendrogram and adjacency matrix visualizations are inherently linear in the presentation of relationships between clusters. A cluster with strong associations to more than 2 clusters will not be able to be adjacent to the third, and so the association to elements of the third cluster appear as "noise," i.e., non-zero entries away from the diagonal.

With the domain of Information Visualization papers, it is not surprising to have visible "+" signs appearing around terms common to the entire domain. Keywords such as 'information visualization' and 'data visualization' are easily identified in the adjacency matrix. In a higher dimension visualization, this these terms may appear as the "gravity" that pulls several other clusters together.

For the Information Visualization keyword domain, this noise affect causes an interesting result that the terms frequently do not seem to cluster together the way that a dictionary or thesaurus would place them. There are many keywords that begin with "*3D*," but they are not clustered together. Likewise, "*Database*" and "*query*" are in different clusters, and *'information visualization'* and *'information visualisation'* (English spelling) are not adjacent.

The clustered adjacency matrix and dendrogram were able to successfully show clusters in the data, but the domain of keywords, even filtered, was too large and sparsely related to generate many meaningful insights. The ability to zoom into the visualizations and pan across them, while they remained linked by keyword is useful. The code was not optimized, however, so re-rendering the visualizations is noticeably slow.

Perhaps the most useful feature in the application is the ability to modify the maximum cluster distance and minimum cluster size, and then zoom into the adjacency matrix and dendrogram to closely inspect the resulting clusters. This visualization allows users to identify true clusters from bimodal and multi-modal imposters. A mere mathematical clustering calculation would not enable this insight.

6. APPLYING THE TOOL TO A SECOND DATA SET

A data visualization tool that is only used to visualize data visualization research is vaguely analogous to a self-licking ice cream cone. Applying ConceptMap to the information visualization research domain was primarily done for the convenience of access to the high dimensional source data. To show the tool's value in other domains, the following paragraphs and images show the result of applying the tool on a popular contemporary concern in mid-2007. The source data analyzed is that of the voting records of United States Senators during the first half of the first session of the 110th Congress. In sum, the source data is 247 roll call votes taken between January 2007 and mid-July 07 [13]

The Senator's voting records are recorded as a series of yes/no/abstain values across a multitude of issues, the similarity between senators is able to be displayed as a gradient range of bright green to dark red, obviously representing the overall vote agreement between 2 senators as green, and disagreement as red. The corpus of information visualization produces a research keywords adjacency matrix that is sparse, resulting in a visualization

with clusters of related concepts identifiable along the diagonal, but mostly white space in the upper and lower triangular matrices.

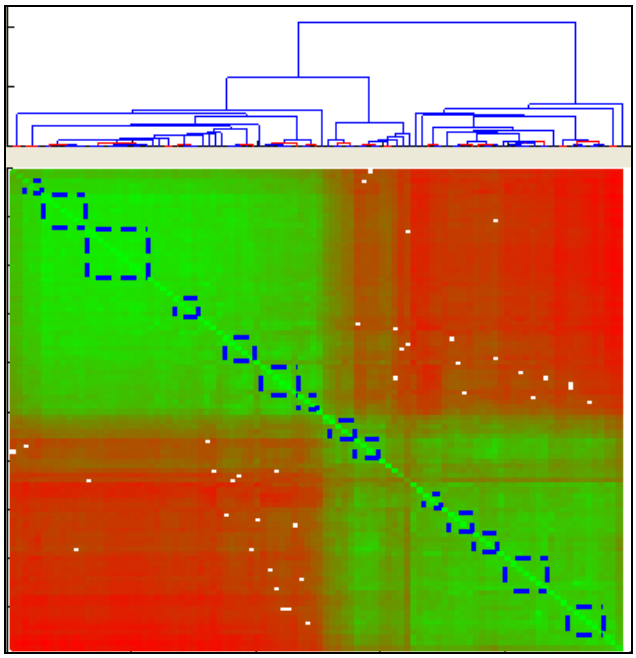


Figure 4. Dendrogram and Adjacency Matrix of the entire Senate, showing clusters of Senators with a minimum cluster distance.

The collection of Senate roll calls, however, is nearly completely populated, with most objects (senators) voting on most roll calls. Rather than white space, the adjacency matrix is a fully colored adjacency matrix providing a heat map visualization of the overall agreement and disagreement of Senator's votes. Figure 4 above shows the entire Senate, and clusters of Senators with a minimum cluster distance. In this way, small blocks of Senators who voted almost identically on every issue are highlighted.

During this time frame, the Republican party held a slight majority of the seats congress. Without labeling the graph, the major party lines could be guessed as the topmost division of the set of Senators, with the majority represented by the cluster on the left because it is larger. The broadly green rectangles reinforce this view, revealing where the majority of Senators from one party tend to vote together on issues. The red rectangles off of the diagonal show that the block of Senators from one party generally disagree with the block of Senators from the other party.

The interesting result from this graphic, however, is that the topmost division of clusters does not break on party lines. The larger grouping on the left includes a block of 14 Republican senators who, by their votes, grouped together, and voted more in line with the Democratic Senators than with their fellow Republicans. This block included (in cluster order) Senator-Snowe-ME-R, Senator-Collins-ME-R, Senator-Specter-PA-R, Senator-Smith-OR-R, Senator-Coleman-MN-R, Senator-Warner-VA-R, Senator-Domenici-NM-R, Senator-Murkowski-AK-R, Senator-Stevens-AK-R, Senator-Lugar-IN-R, Senator-Voinovich-OH-R, Senator-Hagel-NE-R, Senator-Brownback-KS-R, and Senator-McCain-AZ-R. In the graphic below, the crosshairs are highlighting the most Republican-like Democrat (Senator Nelson from Nebraska). The cluster of 14 Republican senators is visible as the right-most grouping after Senator Nelson. The dendrogram and shading of this group show that it is a bi-modal cluster, with Senators Snowe, Collins, and Specter distinctly more Democrat party favoring than the other 11.

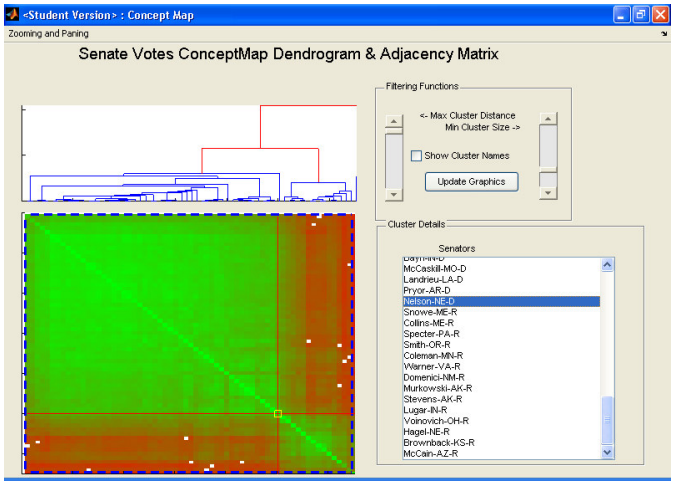


Figure 5. ConceptMap Interface zooming in on the leftmost cluster.

7. FUTURE WORK

Given the ability to continue developing this line of research, the flowing enhancements to the dendrogram and adjacency matrix visualizations would be pursued:

- Adding a legend to define the meaning of the colors and gradients in each of the visualizations.

- Improving the way individual nodes are labeled in the matrix. Currently, the keyword listbox provides the names of the keywords in the map or cluster. Clicking on a keyword causes its location to be highlighted, but the keyword name does not appear in the matrix.

- Enable a search feature to look for keywords without scrolling down the alphabetical list.

- In the cluster focus view, enable a method to jump from one cluster to the next.

- In addition to the sliders for minimum cluster size and maximum distance, add a slider allowing users to adjust a parameter for the "minimum support." This enhancement was recommended after seeing the sparseness of the adjacency matrix and quantity of single associations between elements of a cluster.

- Currently the cluster is named by the leftmost/topmost element contained within it. Frequently, none of the element names are adequate for naming the group, so a recommendation was received to use the default naming that currently exists, but allow the user to rename the cluster.

REFERENCES

[1] Ghoniem, Mohammad; Fekete, Jean-Daniel; and Castagliola, Philippe. "A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations." Proceedings of the IEEE Symposium on Information Visualization, p.17-24. Oct 2004.

[2] H. Siirtola. "Interaction with the Reorderable Matrix." In Proceedings of the Information Visualization '99, London, IEEE, Jul 1999.

[3] Erkki Mäkinen and Harri Siirtola. "The Barycenter Heuristic and the Reorderable Matrix." Informatica . v29 i3. 357-363. Aug 2005.

[4] Henry, Nathalie and Fekete, Jean-Daniel. "MatrixExplorer: a Dual-Representation System to Explore Social Networks." IEEE Transactions on Visualization and Computer Graphics, v.12 n.5, p.677-684, Sep 2006.

[5] Seo, Jinwook and Shneiderman, Ben. "Interactive Exploration of Multidimensional Microarray Data: Scatterplot Ordering, Gene Ontology Browser, and Profile Search." HCIL-2003-25, CS-TR-4486, UMIACS-TR-2003-55. 2003. http://www.cs.umd.edu/hcil/multi-cluster/hce3.html

[6] Borgatti, Stephen P. "How to Explain Hierarchical Clustering." Connections 17(2):78-80. 1994. http://www.analytictech.com/networks/hiclus.htm

[7] Kang, S. "Keyword-based document clustering." In *Proceedings of the Sixth international Workshop on information Retrieval with Asian Languages - Volume 11* (Sapporo, Japan, July 07 - 07, 2003). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 132-137. 2003. DOI= http://dx.doi.org/10.3115/1118935.1118952

[8] Parsons, L., Haque, E., and Liu, H. Subspace clustering for high dimensional data: a review. *SIGKDD Explor. Newsl.* 6, 1. p90-105. Jun 2004. DOI= http://doi.acm.org/10.1145/1007730.1007731

[9] Hinneburg, A., Keim, D. A., and Wawryniuk, M. "HD-Eye: Visual Mining of High-Dimensional Data." IEEE Comput. Graph. Appl. 19, 5, p22-31. Sep 1999. DOI= http://dx.doi.org/10.1109/38.788795

[10] Paley, W. B. "TextArc: Showing Word Frequency and Distribution in Text." Poster presented at IEEE Symposium on Information Visualization. 2002. http://www.textarc.org/

[11] Heer, J. and Boyd, D. "Vizster: Visualizing Online Social Networks." In Proceedings of the Proceedings of the 2005 IEEE Symposium on information Visualization. INFOVIS. IEEE Computer Society, Washington, DC. Oct 2005. DOI= http://dx.doi.org/10.1109/INFOVIS.2005.39

[12] Fekete, J.-D., Grinstein, G., Plaisant, C., IEEE InfoVis 2004 Contest, The History of InfoVis. 2004. http://www.cs.umd.edu/hcil/iv04contest

[13] U.S. Senate Roll Call Votes 110th Congress - 1st Session (2007). http://www.senate.gov/legislative/LIS/roll_call_lists/vote_menu_110_1.htm. Accessed 15 July 1997