A Node Aggregation Strategy to Reduce Complexity of Network Visualization using Semantic Substrates

Aleks Aris and Ben Shneiderman

Human-Computer Interaction Lab
University of Maryland, College Park

ABSTRACT

Semantic substrates are spatial templates for networks, where nodes are grouped into regions and laid out within each region according to one or more node attributes. Analysts' ability to design their own substrates leads to a different approach than other more automatic approaches to layout nodes (force-directed, circular, etc.). While the semantic substrate approach provides more interpretable node locations, sometimes a set of nodes is compressed into a small space on the display, leading to node overlap. In this paper, we significantly improve this situation by introducing the node aggregation mechanism in the context of semantic substrates. We illustrate this functionality in a document citation network and provide pros and cons of the approach. We conclude with guidelines and future directions for this research. Throughout the paper, examples are illustrated with NVSS 3.0, the network visualization tool developed to explore the semantic substrate idea.

KEYWORDS: Network visualization, semantic substrate design, information visualization, data exploration and analysis, node aggregation, clustering, graphical user interfaces.

INDEX TERMS: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Graphical User Interfaces (GUI); H.5.2 [Information Interfaces and Presentation]: User Interfaces—Screen design (e.g., text, graphics, color)

1 INTRODUCTION

Networks from a diverse set of domains have lead to numerous visualizations [9] to support the tasks of domain experts that include detecting clusters, identifying interesting nodes, and finding interesting relationships between nodes through link patterns ([6], [7] provide a taxonomy of such tasks). Many of these visualizations focused on dealing with the presentation of node-link diagrams following several guidelines, such as minimizing link crossings and maximizing symmetry, commonly known as graph drawing aesthetics ([11], [14]). However, giving these guidelines the highest priority leads to arbitrary location of nodes on the display. To improve the comprehensibility through the goal of interpretable node location lead to the semantic substrate idea. The benefits of semantic substrates were illustrated in the legal precedent domain [6]. Furthermore, guidelines on how to design semantic substrates were reported by illustrating them in two case studies [19].

Semantic substrates, first, group nodes into rectangular regions, and then lay them out within each region according to user-chosen node attributes. Link visibility is controlled by source and destination regions and attributes used to place nodes within regions.

To reduce complexity in dense networks, we introduce node aggregation in semantic substrates. This powerful addition, when combined with the filtering and details-on-demand functions enables analysts to detect patterns, gaps, outliers, and clusters in large datasets.

In our strategy, node aggregation is based on replacing all the nodes in a grid cell with a single metanode. Grid cells are the result of using the GridPlotXY placement method of NVSS ([19]), where x- and y-axes are used and they each represent the values of a node attribute. In this paper, we illustrate node aggregation in the context of the GridPlotXY placement method; however, it could be generalized to other placement methods. Analysts can easily switch between nodes and metanodes. In the nodes mode, all nodes are displayed. In the metanodes mode, nodes in grid cells are aggregated into a single large metanode. The paper illustrates how these modes are useful in scaling up to explore much larger datasets than was possible in our earlier work. Our broadly applicable strategy depends only on aggregating nodes with similar attribute values, avoiding costly clustering algorithms.

Section 2 reviews relevant work to node aggregation. Section 3 illustrates the process of exploration of an example dataset in the context of node aggregation. Section 4 offers guidelines for node aggregation and gives examples based on section 3. Section 5 details future work and section 6 concludes the paper.

2 RELEVANT WORK

As the number of nodes and links increase, meaningful network exploration becomes more challenging for analysts. While larger displays are modestly helpful, several strategies have been employed to reduce the complexity of networks such as: node aggregation, focus+context techniques, link aggregation and routing, graph drawing aesthetics, and matrix-based representations.

Among the node aggregation approaches, PivotGraph [1] and Jambalaya [5] show the closest characteristics to our work in semantic substrates. PivotGraph [1] "rolls-up" a graph to produce a summary view of nodes and links on a two-dimensional grid based on a node attribute on each of the axes, which is similar to node aggregation in one of the regions in NVSS. While PivotGraph allows analysts to change the node attribute on the axes and animates the transition, NVSS allows analysts to change the node attributes via its Substrate Designer interface. PivotGraph also uses size coding for metalinks (besides metanodes). Jambalaya [5] uses a graph metaphor to show connections between concepts via links. Concepts are similar to regions in NVSS. Concepts can contain sub-concepts, which resembles node aggregation (concepts to sub-concepts are similar to metanodes to nodes). Nodes can be placed manually or automatically by a structural property of nodes but not by a node

Street Address and Electronic Mail Address

attribute. Links can be color-coded according to their types, which is determined by their source and target classes.

Several approaches use multiscale visualization ([24], [21], [12]). Such applications usually are able to handle very large graphs and provide analysts to group nodes to metanodes and ungroup them. In Grouse [21], nodes are grouped into metanodes using hierarchy information and the layout is based on topological features that are computed from the graph structure. Analysts are able to open and close metanodes on demand and layouts are computed as needed as the user explores the network, which enables fast response for very large networks. Tulip [12] enables analysts to manage clusters (group and ungroup nodes) as well, and although it supports node attributes, there is no layout strategies using node attributes. However, a plug-in capability allows defining new layout algorithms. Link visibility based on node attributes also has not been reported. SocialAction [2], designed primarily for social networks, uses a force-directed layout and visually surrounds clusters of nodes with a convex hull. The clusters are determined using hierarchical clustering, with interactive parameter control by analysts, on the network's structure and not the node's attributes. Each cluster can be collapsed to a metanode where the links become metalinks. Metanode size and metalink thickness represent the number of nodes and links they represent.

Several applications do not reduce nodes into metanodes but use visual clustering and provide filters to cope with the complexity. NicheWorks [3] clusters nodes by placing them close to each other and doesn't reduce them to metanodes. It uses an initial layout (circular layout, hexagonal grid, and tree layout) and through incremental algorithms (e.g. steepest descent and simulated annealing) computes the final layout. NicheWorks was designed to visualize large graphs (up to 1,000,000 nodes) and and supports filtering based on node attributes. Osprey [13], a domain specific tool for biological researchers, also handles large datasets and provides node filters based on attributes. However, it doesn't provide layouts based on node attributes and doesn't reduce nodes into metanodes.

The use of hyperbolic geometry in networks ([22], [23]) exemplifies focus+context techniques to cope with complexity. Although they provide greater detail for the focused areas, they may distort the overall view of the network, which leads to difficult navigation. These techniques do not aggregate nodes.

Another strategy to reduce complexity in networks involves links. Becker et al. [4] reduces the display of double directed links between two nodes to a color- and thickness- coded straight line. Another method is to draw part of the links. Flow Map Layouts [16] reduce the amount of links by combining common parts via edge-routing algorithms. Other techniques to organize links include aggregating them using node hierarchy information [10], which improves link display.

The application of graph drawing aesthetics [11], [14] could be considered as another strategy to reduce complexity of the network visualization. Principles, such as minimization of link crossings, organize the display to reduce the complexity. However, these techniques seem to place nodes arbitrarily and not according to node attributes.

Matrix-based approaches (Ghoniem et al. [18], NodeTrix [17]) provide an alternate view to the node-link diagrams. They reduce display complexity by avoiding drawing nodes and links in traditional ways. In such representations, the spatial structure of the network is hidden. NodeTrix combines matrices with node-link diagrams to give a sense of the spatial structure (not based on node attributes) and simplifies the display by keeping matrices for parts of the network. In addition, interactive operations, such as sorting matrix columns in terms of attributes and filtering, are provided.

3 EXAMPLE: TOBIG DATA

This section shows how node aggregation helps analysts explore a document citation dataset. We designed the substrates in consultation with our collaborators. Our main collaborators are Prof. Noshir Contractor at Northwestern University and Assistant Prof. Steve Harper at James Madison University. The dataset has Tobacco researchers, the documents they wrote, and the keywords used in these documents. Our collaborators have been interested to find patterns and relationships to answer several questions including what topics emerged over time in this field; when, how, and by whom they emerged; what the expertise of authors is, and in which areas.

Nodes represent authors, documents, and keywords. There are 29 authors, 1,700 documents, and 2,567 keywords, totaling to 4,296 nodes. Links in the dataset are directed and represent the following relationships according to their source and destination: Authors write documents, documents cite documents, and documents use keywords. There are attributes that depend on the "type" of nodes. Those attributes are present for the other types of nodes (as NVSS only accommodates nodes having the same set of attributes) and they are conveniently ignored where they are non-applicable (which is a work-around and for NVSS to support different types of nodes could be considered as future work). The *type* attribute has values "Author," "Document," or "Keyword." The *name* attribute is the name for authors, the title for documents, and the name for keywords. The *year* attribute is the first year of publication for authors, the year of publication for documents, and the year of the first document that it appeared for keywords. The attribute *CR* is only applicable to authors; it stands for "Citations Received" and it actually represents the author's H-score (an index to characterize the scientific output of a researcher, where the researcher has h papers that are cited h or more times, [20]). The attribute *LCS* is only applicable to documents; it stands for "Local Citation Score" and it represents the number of times the document was cited by other documents in this dataset. *Count* is only applicable to keywords and it represents the number of appearance of this keyword in documents.

The semantic substrate in Figure 1 has three regions, each using a value of the *type* attribute. The location of the regions from top to bottom is in line with the directionality of the links: authors write documents and documents use keywords. *Year* is used along the x-axis of all regions consistently. *CR, LCS,* and *Count* are used along the y-axes with a consistent bin height, 5 for authors and 10 for documents and keywords. Most nodes seem to have lower values and tend to overlap in small places. The dataset is dense in terms of the links as there are 1,770 author-to-document, 4,966 document-to-document, and 9,649 document-to-keyword links (total = 16,385). The MODE section has modes of aggregation: "Nodes" and "Metanodes."

Node overlaps in lower values of *LCS* and *Count* suggests that it might be better to have another substrate that gives more space to the lower values. The y-axis of regions in Figure 2 is manually created in the Substrate Designer of NVSS to support conceptually distinct categories for each node type (For strategies to create uneven distributions see Aris et al. [8]). For example, authors with H-score between 5 and 9 are treated similarly, so are allotted the same horizontal band.

When compared with Figure 1, Figure 2 has a better spread of nodes although there are still cells that have more nodes than the available space. Keywords having *Count* = 1 are such cells. In addition, the number of links exceeds the limit for a comprehensible display. Figure 3 (left) shows all 9,649 document-to-keyword links.

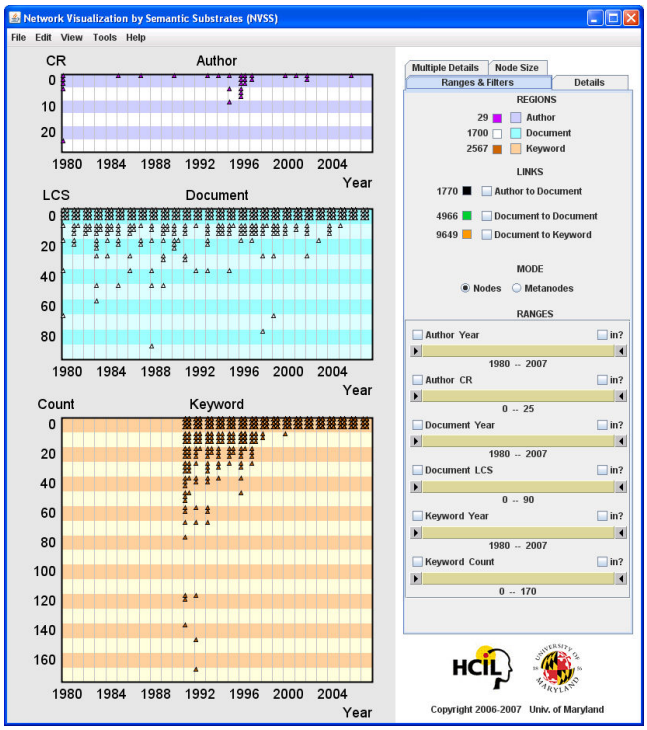


Figure 1 An initial semantic substrate is applied to the ToBIG dataset, where nodes represent authors, documents, or keywords, and links represent "Author writes Document", "Document cites Document", or "Document uses Keyword". Nodes are grouped into regions using the type attribute with "Author", "Document", and "Keyword" values while they are placed using Year along the x-axis and CR, LCS, and Count along the y-axes.

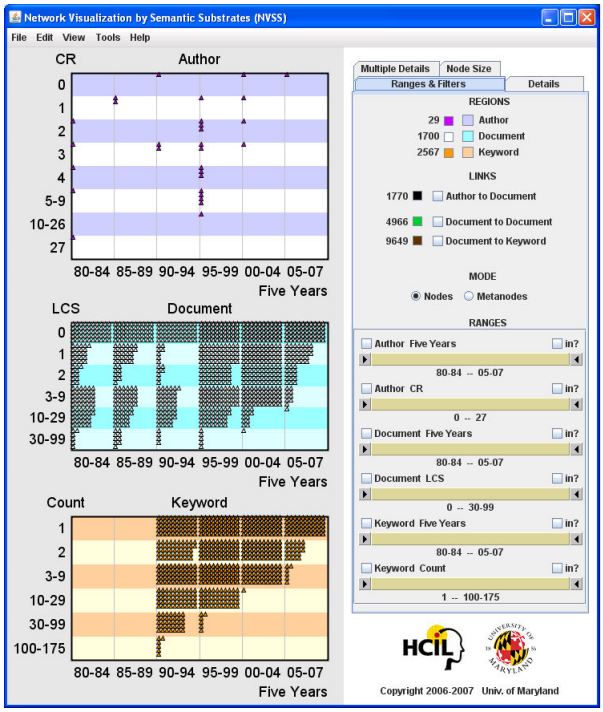


Figure 2 A different substrate is applied to the data in Figure 1 upon seeing node overlap in lower y-values. Year on the x-axis consistently binned into 5-year periods, while a custom binning is applied for CR, LCS, and Count on the y-axes different for each region.

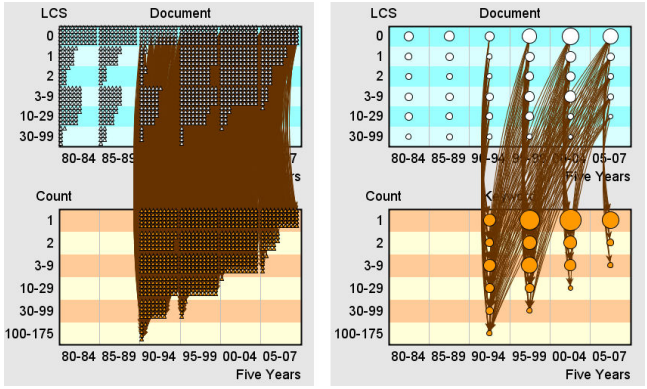


Figure 3 Due to their large number (9,649), the document-to-keyword links are beyond the limit of comprehension (left). In the metanodes mode (right), where nodes in cells are aggregated to metanodes, the display becomes simpler and more comprehensible. Metanode size indicates the number of aggregated nodes.

Under the MODE section on the control panel (Figure 2), analysts click "Metanodes" to aggregate nodes. The nodes in each cell are aggregated to a metanode. Links between metanodes become aggregated, as well. There is a link between a metanode and another node if there is a link between at least one of the children of the metanode and that other node. The node display becomes much simpler and the links become more comprehensible (Figure 3, right).

Looking at Figure 3 (right), it seems that all or almost all possible links are present. Analysts can further investigate by applying filters (Figure 4).

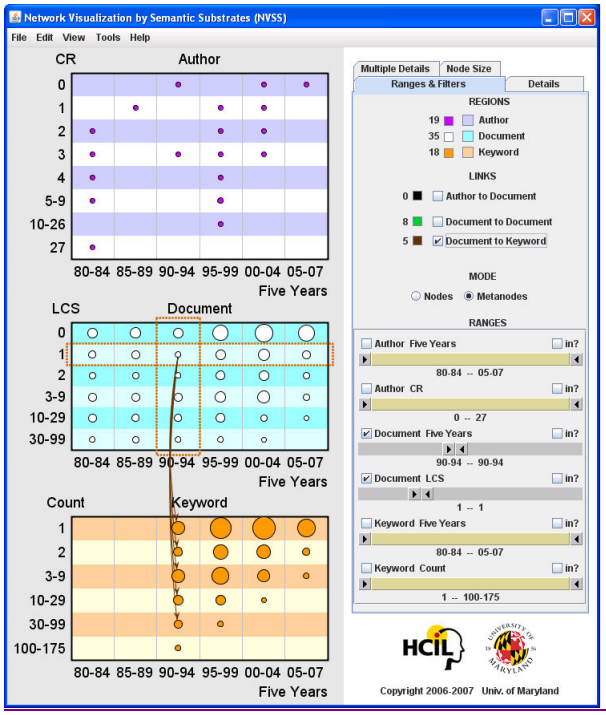


Figure 4 Restricting the links in Figure 3 (right) to the 90-94 range and looking at the documents that are cited once in the dataset. Interestingly, none of them uses any of the most highly used keywords.

By using both the *LCS* and the *Five Years* filters on the Document region, analysts restrict outgoing links to the most cited documents in the 90-94 year range. By looking at the document-to-keyword links, analysts can quickly iterate through all the 6 categories in the Document region to see the usage patterns of Keywords. Figure 4 shows the links from documents that have been cited once in the dataset. Interestingly, none of these documents have used any of the most highly used Keywords introduced in the same *Five Years* bin. Upon noticing this interesting fact, analysts can quickly browse the other 5 categories in the Document region within the same *Five Years* bin to see whether this was also true with the other groups. Since space is limited in this paper, we show only 2 categories (Figure 4 and Figure 5). Analysts find that most of the other categories use the most highly cited Keywords (specifically with *Count* = 0, 2, 3-9, 10-29) except one other category, which is the 30-99 (Figure 5). The link from this category of documents to the most used keywords is missing! In other words, this interesting fact emerges: the most cited keyword introduced in the same 5-year period has *not* been used by any of the most cited documents!

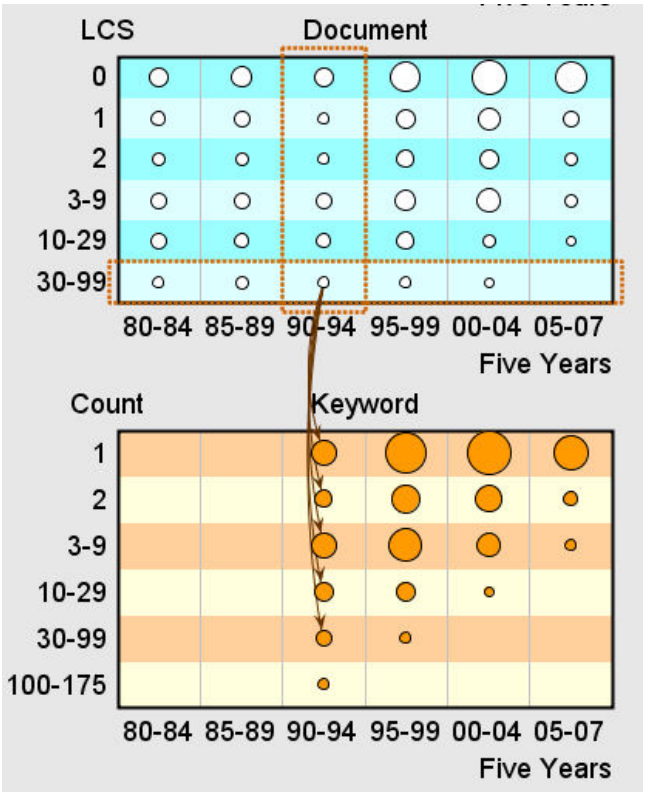
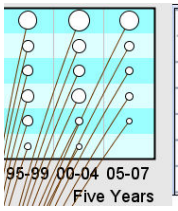


Figure 5 Iterating over the other categories in the Document region within the same Five Years bin (90-94) to see the citation patterns reveals even a more interesting fact than the one in Figure 4: The most highly used keywords are not used by any of the most cited documents within the same 5-year period!

Clicking on the metanode representing the most highly used keywords gives a list on the control panel (Figure 6, the table on the right hand side). Putting the link filters only in the Keyword region around the most used keywords reveals that in all the upcoming 5-year periods, all categories of documents use at least one of these keywords (Figure 6, left hand side, notice how every metanode in the Document region is linked).



ID	NAME	TYPE	YEAR	CR_LCS_C
5382	tobacco	Keyword	1992	152
5264	study	Keyword	1992	175
3423	cigarette	Keyword	1991	121
4559	nicotine	Keyword	1991	149
5190	smoking	Keyword	1991	129
5189	smoker	Keyword	1992	123

Figure 6 Putting filters around the most highly used keywords and looking at the documents, we see that all documents after the 90-94 period (see left) use at least one of the highly used keywords (see keywords on the right).

One question about this dataset is the relationship between authors and documents. What type of author writes the mostly cited documents? Considering that the documents cited 10 or more times are the top documents in this dataset, filters could restrict the incoming links to documents that have *LCS* >= 10 (Figure 7).

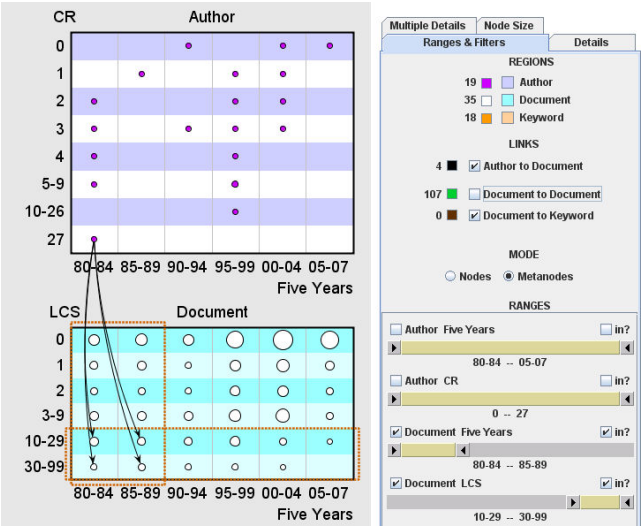


Figure 7 Looking at the top documents in the 80-89 period. They are written only by the top author.

It turns out that the top documents during 80-89 are written exactly by one category of authors (Figure 7). Switching to the nodes mode reveals that there is exactly one such author: Steve Hecht. He has the highest H-score (27) and began to write in 1980 (this information is available through the Details tab in the control panel). He can be considered as the leader in the 80s period. Switching the view to the nodes mode reveals that he wrote a total of 64 top documents in the 80s (the number 64 appears next to the "Author to Document" links; not illustrated). Next, we look at the 90-94 period (Figure 8, left).

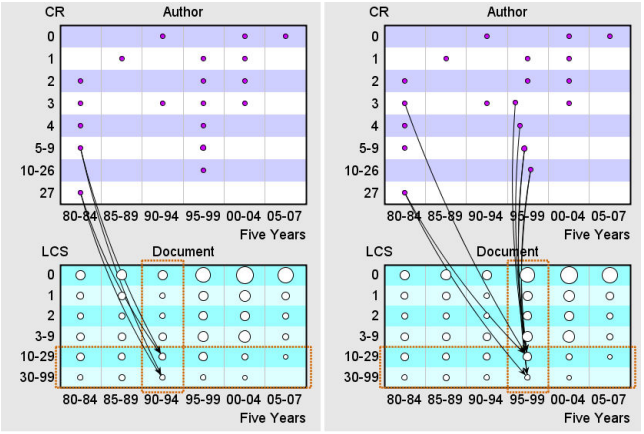


Figure 8 Left: Looking at the top documents in the 90-94 period. Another category of authors (5-9) joins the one that has H-score 27. Right: Looking at the top documents in the 95-99 period. Nodes are manually displaced for better link display.

In the 90-94 period, Steve Hecht (*CR* = 27) continues to write top papers while another category of authors (those who have *CR* 5-9) join him. The nodes mode shows the category has only one author, namely David Ashley (year = 1980, H-score = 6). Both authors have papers in both sections (10-29 and 30-99).

Looking at the next 5-year period (Figure 8, right) reveals that 5 new categories of authors start writing top documents. Steve Hecht (H-score = 27) continues to contribute while David Ashley (*CR* = 5-9) does not any more.

Among these 5 new categories, only one of them started writing in the 80s. In fact, he is Richard Clayton (year = 1980, H-score = 3). All the other 4 category of authors started writing in the same 5-year period. (There are 10 authors in those 4 categories. In increasing H-score from 3 to 26, there are 2, 3, 4, and 1 author(s), respectively.) Nodes are manually displaced for better link display (Figure 8, right).

In the next 5-year period (2000-2004, see Figure 9, left), Steve Hecht (H-score = 27) and the top 3 categories of authors that began writing in the previous 5-year period (95-99) continue to write top documents. This is with the realization that the measure for top documents does not have the same level of difficulty across the 5-year periods. Since 2000-2004 is more recent, it is harder to produce documents that are cited 10 or more times. In fact, there are only 16 such documents. Shifting the threshold for *CR* to include 3-9 reveals that the authors that have written top documents in the previous time periods are active and writing documents in the 3-9 range in addition to other newly contributing authors (not illustrated).

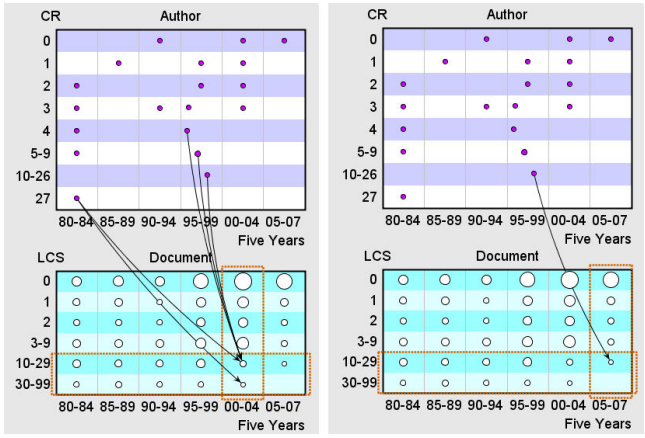


Figure 9 Left: Looking at the top documents in the 2000-2004 period. Nodes are manually displaced for better display. Right: Looking at the top documents in the 2005-2007 period.

Looking at the last period (2005-2007, Figure 9, right), we realize that only one category of authors write these. Switching to the nodes mode (not illustrated) shows that the category contains only one author (Neal Benowitz, year = 1995, H-score = 10) and there is only one document written (by him) in the 2005-2007 period (written in 2005 and cited 13 times). Realizing that the 05-07 bin represents a 3-year period as well as being the most recent period helps explain this.

This exploration over the years gives a quick sense of what type and sometimes which authors wrote the top documents over time. For instance, Steve Hecht (with H-score = 27) has remained active over the years consistently contributing to the top documents (except the last 3 years). In fact, more is revealed when looking at the pattern of his documents (Figure 10, metanodes are size-coded based on the number of nodes they represent). He has written almost all types of documents (in terms of year and citation score), specifically 34 out of 35. Metanode size in the last category, which he did not contribute, indicates that this category has a small number of documents, anyway. The nodes mode shows that he has written 556 documents in total. It is interesting to see that he did not only write top documents but all types of documents including 203 documents that never got cited in the dataset. He has written documents with a fair distribution in terms of being cited (58, 46, 125, 101, 23 documents that have LCS 1, 2,

3-9, 10-29, and 30-99, respectively). He has written a total of 124 of the 152 top documents in the dataset. If he is considered successful, perhaps, one way to become successful is to be as prolific as he was and keep writing over the years without too much regard to whether a document is cited or not. Apparently, he continued to write even though many of his documents did not get cited. In addition, he was the pioneer of the 80s and produced about half of his top documents at that time (64 of 124).

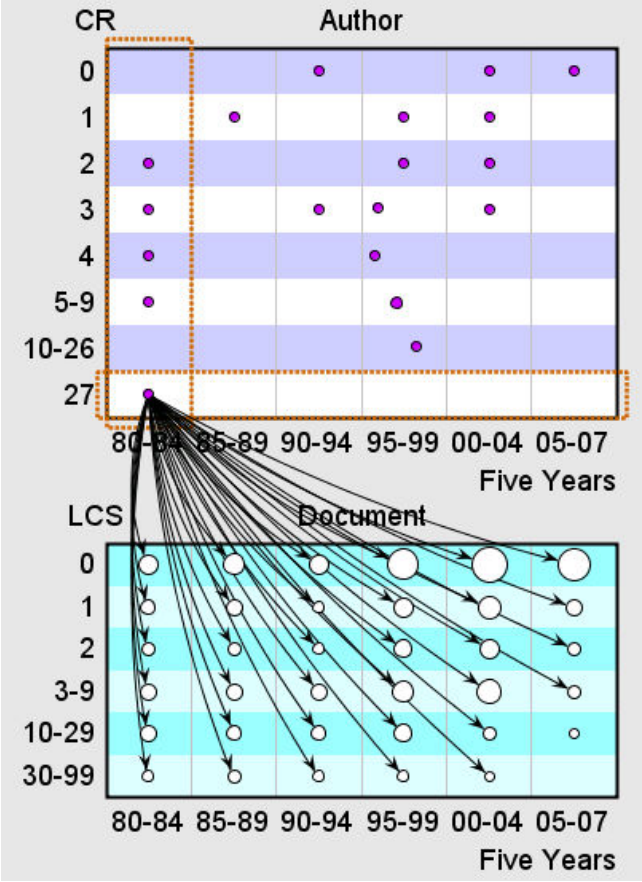


Figure 10 Documents that Steve Hecht (H-score=27) wrote. Metanodes are size-coded by the number of nodes they represent.

The exploration in the ToBIG dataset illustrates how node aggregation simplifies discovery and helps analysts find interesting facts, such as the most used keywords were not used by the most cited documents immediately. Node aggregation also helps analysts get an overview, for example, what type of authors have written documents and what the characteristics of those authors are with the option of detail on demand (author names in the categories explored, how many there were, how many papers they wrote, etc.). Analysts can gain an in-depth understanding of one aspect of the dataset (Steve Hecht's profile in terms of writing papers), which has the potential to produce questions and/or hypotheses (Can we consider Steve Hecht as successful? Assuming so, perhaps the key to a success similar to his is to start early, write consistently, and accept the fact that not all documents will be highly cited.).

4 Design Choices

There could be several possible ways to do node aggregation in the context of semantic substrates. We considered th following alternatives:

(1) Aggregate to several metanodes in each cell. Aggregating nodes to several metanodes in each cell could have the advantage of providing more detail to analysts with the aim to accommodate as much as detail as possible (i.e. maximum use of available space). However, several issues with this approach convinced us that this alternative is less favorable. One issue is the question of how many metanodes should it be aggregated to and how to select the nodes each metanode represents. This issue becomes rather complex quickly, which might be worthy of further exploration; however, we believe that conveying the meaning of the specific aggregation would be difficult. The confined space of a cell is likely to make the links within the cell hard to comprehend, which is only incrementally advantageous over the non-aggregated form. Hence, we decided to limit our scope to one metanode per cell.

(2) Support a "mixed" mode, where nodes in a cell are aggregated to a metanode only when they do not fit the available cell space. We implemented this approach and had several exploration phases, where we quickly discovered that the fact that some cells are aggregated and some aren't was too disorienting for analysts. In addition, this mixed mode does not support analysis tasks well, as tasks tend to fit either when all nodes are aggregated or non-aggregated; especially they do not usually involve the component of "if the nodes do not fit the available cell." Hence, we did not further explore this direction where this mode is supported.

5 Guidelines

The following sections discuss how to apply node aggregation in the context of semantic substrates.

5.1 Simplified exploration through node aggregation

Node aggregation helps to attain a more comprehensible display and also facilitates understanding by simplifying the display. When analysts select meaningful attribute values to group nodes, aggregated nodes become meaningful overviews of the groupings the user made. The simplicity makes the exploration effective and efficient. Facts stand out, especially surprising ones. The lack of use of the most used keyword within the same 5-year period in Figure 5 and the fact that Steve Hecht wrote all types of documents (Figure 10) are examples.

One substrate is better than another in answering a specific question or exploring from certain perspectives of the dataset. Regarding the dataset in this paper, we were interested in the activity (for authors: writing papers, for documents: being cited, and for keywords: being used) and time aspects of nodes (authors, documents, and keywords) as well as the relationship between them. The attributes used in the substrates (*CR, LCS, Count*, and *Five Years*) supported the exploration from the perspectives above.

5.2 Binning attribute values into ranges

The dataset in section 3 looked from the perspective of 5-year periods and *CR, LCS,* and *Count* binned in a certain way. This is a good arrangement when analysts know and want to see the data in this way. In other words, it makes sense to look at the data in those specific 5-year periods and in the *CR, LCS, and Count* ranges that were used. For example, in terms of analysts' understanding, the local citation score (LCS) of a document does not make much difference within the range 5-9; hence, the range of 5-9 is given a specific slot and separated from other ranges.

Similarly, there is (or at least could be) a difference when LCS is 3 and 4; hence, the different slots were allotted.

There is a trade-off in how to bin values into ranges. The more bins, the more detailed information revealed, and the more effort needed in managing it (remembering and comparing them to each other). On the other hand, too few bins lead to a crude, and therefore, too shallow an understanding. A balanced view is desired and can be attained by iterative substrate design (see also Aris et al. [19]). In the example dataset, a 6-part binning for the documents and a 7-part binning for the authors is used. Figure 1 shows one of the earlier substrates on the same dataset. The latter binning arrangement arose after perusing the distribution of the data through a few iterations. A certain amount of time may be necessary to achieve a satisfactory result. We believe this depends on many factors, such as the complexity of the dataset, how much analysts know about the dataset, and how experienced they are in terms of having explored the dataset.

Certain tasks are better with certain substrates (and binning) than others. In the example in this paper, the latter substrate performed well in terms of providing insights and understanding to the data. The 6-7 bins on both axes made it optimal to go over the different slots and get overviews quickly as well as compare them to one another. If deeper or other types of questions arise, substrates could be iteratively modified to look for deeper insights and more precise facts.

5.3 Details-on-demand

Being able to switch between the metanodes and nodes modes allowed looking at details-on-demand. This way, analysts get more information only when needed, which leads to a cleaner, and therefore, a more comprehensible and efficient process of exploration.

Details-on-demand have several benefits: They (1) enrich understanding due to the additional information, (2) help to check assumptions, and/or (3) prevent incorrect inferences and sometimes compensate for when the representation of the overview is misleading.

Examples for the above points are as follows:

(1) In Figure 8 (right), switching to the nodes mode revealed that there are 10 authors in the 2000-2004 period and what their distribution is in terms of H-score. Another example is the statistics about Steve Hecht: that he wrote 556 documents, 124 of which were top documents (out of 152 top documents) and that in 1980s, he wrote about half of those (64 top documents).

(2) In Figure 9 (left), it is assumed that it is harder to write top-documents in the 2000-2004 period, as it is a recent period. Switching to the nodes mode shows that this assumption is correct as there are only 12 top documents while in earlier 5-year periods their number is at least 22.

(3) In Figure 9 (right), looking at the nodes mode reveals that there is only one top document in the 2005-2007 period, which is written by only one author. This prevents treating this last period the same as (or close to) the previous ones as there is substantial difference.

5.4 Performance & Scalability

NVSS provides smooth interaction up to 1000-2000 nodes on a 3GHz Dell 8400 with 3GB RAM. Larger networks that reduce to this size in the metanodes mode could be smoothly explored. The transition time from and to metanodes mode is 2 seconds for the example dataset in this paper. For another dataset with 10,000 nodes and 17,836 links, the transition time from nodes to metanodes is approximately 2 seconds. The transition time from metanodes to nodes is 7-8 seconds. We also tried NVSS with a dataset containing 29,555 nodes and 352,807 links. The transition time from nodes to metanodes is 70 seconds and the transition from metanodes to nodes is 155 seconds. In general, the transition times depend on the network size (number of nodes and links), the substrate applied (how nodes are distributed among regions and cells), and perhaps slightly on the structure of the network.

6 Future Work

One possible future work would be to assist analysts in binning attribute values into bins. One way to do this would be to implement a visual module that shows the distribution of the attribute values and suggests binning intervals to sustain a balanced distribution of nodes and links. It would be beneficial to provide analysts to adjust the suggested bin intervals to facilitate the bin creation for many custom bin intervals analysts would like to have. A sophisticated interface would be capable of providing several alternatives by conveying to the user the trade-offs in each alternative. This way analysts could make informed decisions (for a similar idea in forecasting time-series interfaces, see [26]).

Exploration of the data usually involves filtering and narrowing down to an interesting subset. Node aggregation provides overviews for improved understanding. Assuming that two links are pointing (A and B) to an aggregated node, via filtering there are situations that only A is visible. However, the aggregated node remains the same. It has the meaning "some nodes that the aggregated node represents are linked." A way that gives information about the nodes that are linked would be helpful. One way to do this is to have two types of aggregated nodes, one that represents the linked nodes and the other the rest. This way, analysts can set the size coding in the Substrate Designer to represent any attributes that they want to be reflected onto the visualization.

Semantic substrates involve regions and subsets of regions through filtering. The support for a dynamic selected set of nodes will enable analysts to do cascaded types of exploration. In other words, if analysts could select nodes linked to (or linked from) after a certain filtering and can use those selected nodes to arrive at other nodes through links, this will enable analysts to arrive at more complex meaningful subsets of the data. For example, in Figure 8 (left), if we could select the documents and then see the set of keywords these documents use, we would have performed a cascaded exploration. The exploration could continue by finding the authors that used those keywords. As such cascaded explorations get longer, it becomes harder to keep track of the meaning of the selected subset. A visual representation that reminds analysts of the meaning is likely to be needed. A history mechanism will enable analysts to undo steps in their cascaded exploration and choose other paths [24]. Being able to compare different paths may add additional benefits. Being able to define more than one dynamic selected set of nodes may expand the types of explorations analysts could do.

Another type of improvement would be the capability to compare two (or more) link patterns arrived through exploration. Currently, this can be achieved by having two instances of the visualization side by side. With additional features, there may be benefits to incorporate more than one exploration view in the same application.

Link display in the metanodes mode could be improved by size, color, or texture-coding to indicate the number of links they represent. In certain tasks, link tasks may be reduced to identifying source and target nodes through some mechanism. In such cases, link rendering can be eliminated for simpler display to facilitate comprehension.

Further future work includes supporting more than one type of node (e.g. bimodal networks), allowing multiple valued attributes, improved performance for scalability, and additional filters for nodes and links, and perhaps widgets for various visual interactions. Different-shaped and overlapping regions might help

in some datasets. Evaluation of semantic substrates in several domains by case studies could also be extended [15].

7 Conclusion

With node aggregation in the context of semantic substrates, we believe that analysts will be able to explore larger datasets faster and more effectively. This paper demonstrated this via an exploration on an example dataset.

The ability to switch between the aggregated view and the non-aggregated view allows analysts to access details as needed. This helps to accelerate the process of exploration by eliminating elements and details that are not of interest. As a result, analysts can better concentrate on the simpler version of the information while seeking facts about the data that they are looking for.

By looking at the current node aggregation feature in NVSS, this paper also conceptualized several aspects of exploration related to node aggregation, such as binning the attribute values, choosing attributes for the substrate for effective exploration, and detail on demand. It also revealed future work that would potentially enhance the quality of the exploration process. These include representing connected nodes to filtered links in aggregate form, ability to define a set of nodes through filtered links to support cascaded exploration, visual coding for aggregated links to represent the actual connections, and further future work including an enlarged scope of network data applicable to semantic substrates.

We believe that node aggregation enhances the benefits of using semantic substrates (which increase user control that lead to better understanding) by enabling simpler, and therefore, a more efficient and effective exploration of networks.

Acknowledgements

We appreciate the invaluable collaboration of Prof. Noshir Contractor (Jane & William White Professor of Behavioral Science at Northwestern University) and his collaborator Assistant Prof. Steven Harper (Management Program, James Madison University). We thank our colleagues that gave us useful comments.

References

[1] M. Wattenberg. Visual Exploration of Multivariate Graphs. *Proc. SIGCHI Conference on Human Factors in Computing Systems*, 811-819, 2006.

[2] Perer, A. and Shneiderman, B., Balancing Systematic and Flexible Exploration of Social Networks, *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 12 (5), 693-700, 2006.

[3] G. Wills. NicheWorks – Interactive Visualization of Very Large Graphs. *Journal of Computational and Graphical Statistics 8*(2), 190-212, 1999.

[4] R.A. Becker, S.G. Eick, and A.R. Wilks, Visualizing Network Data. *IEEE Trans. on Visualization and Computer Graphics 1*(1), 16-28. March 1995.

[5] Storey, M.-A.D., Musen, M.A., Silva, J., Best, C., Ernst, N., Fergerson, R. and Noy, N.F., Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protégé, In *Workshop on Interactive Tools for Knowledge Capture, K-CAP-2001*, (Victoria, B.C. Canada, 2001)., 2001.

[6] Ben Shneiderman, Aleks Aris, Network Visualization by Semantic Substrates, *(Proceedings of IEEE Visualization / Information Visualization) IEEE Trans. on Visualization and Computer Graphics 12*(5), 733-740, 2006.

[7] Bongshin Lee, Catherine Plaisant, Cynthia Sims Parr, Jean-Daniel Fekete, Nathalie Henry, Task taxonomy for graph visualization, *Proc. 2006 AVI workshop on Beyond time and errors: novel evaluation methods for information visualization*, 1-5, 2006.

[8] Aris, A., Shneiderman, B., Plaisant, C., Shmueli, G., Jank, W., Representing Unevenly-Spaced Time Series Data for Visualization and Interactive Exploration, *Proceedings of the International Conference on Human-Computer Interaction (INTERACT 2005)*, LNCS 3585, 835-846, 2005.

[9] Herman,I.,Melançon,G., Marshall, M. S., Graph Visualization and Navigation in Information Visualization: A Survey, *IEEE Trans. on Visualization and Computer Graphics 6*(1) (2000): 24-43.

[10] Danny Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. *IEEE Tran.s on Visualization and Computer Graphics 12*(5): 741-748, 2006.

[11] Colin Ware, Helen Purchase, Linda Colpoys, and Matthew McGill, Cognitive measurements of graph aesthetics. *Information Visualization 1*(2), 103-110, 2002.

[12] D. Auber. Tulip: A huge graph visualisation framework. In P. Mutzel and M. JÃ¼nger, editors, Graph Drawing Softwares, Mathematics and Visualization, 105-126. Springer-Verlag, 2003.

[13] Bobby-Joe Breitkreutz, Chris Stark, and Mike Tyers, Osprey: a network visualization system, *Genome Biology 4*(3): R22, 2003.

[14] G. Sindre, B. Gulla, H. Jokstad, Onion Graphs: Aesthetics and Layout, *Proc. IEEE Symp. on Visual Languages*, 287-291, 1993.

[15] Ben Shneiderman, Catherine Plaisant, Strategies for Evaluating Information Visualization Tools: Multi-dimensional In-depth Long-term Case Studies, *Proc. BELIV'06 workshop, Advanced Visual Interfaces Conference*, 2006, Venice.

[16] Doantham Phan, Ling Xiao, Ron Yeh, Pat Hanrahan, and Terry Winograd, Flow Map Layout, *IEEE Symposium on Information Visualization (INFOVIS 2005)*, 219-224, 2005.

[17] Nathalie Henry, Jean-Daniel Fekete, and Michael J. McGuffin, NodeTrix: A Hybrid Visualization of Social Networks, *IEEE Transactions on Computer Graphics and Visualization (Proceedings Visualization/Information Visualization 2007)* Volume 13, Issue 6 November 2007, pages 1302-1309.

[18] M. Ghoniem, J.-D. Fekete, and P. Castagliola, A Comparison of the Readability of Graphs using Node-Link and Matrix-Based Representations. *Proc. IEEE Symposium of Information Visualization 2004,* 17-24, 2004.

[19] Aris, A., Shneiderman, B., Designing Semantic Substrates for Visual Network Exploration, *Information Visualization Journal 6*(4), 281-300, 2007.

[20] Hirsch, J.E., An index to quantify an individual's scientific research output, *Proceedings of the National Academy of Sciences of the United States of America 102* (46), 16569-16572, 2005.

[21] Daniel Archambault, Tamara Munzner, and David Auber, Grouse: Feature-Based, Steerable Graph Hierarchy Exploration, *Eurographics/IEEE-VGTC Symp. on Visualization*, 67-74, 2007.

[22] John Lamping, Ramana Rao, Peter Pirolli, A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies, *Proc. ACM Conf. Human Factors in Computing Systems, CHI*, 2005.

[23] Tamara Munzner, Drawing Large Graphs with H3Viewer and Site Manager, *Proc. Symp. Graph Drawing'98,* (1998): 384-393.

[24] Auber, D.; Chiricota, Y.; Jourdan, F.; Melancon, G.; Multiscale visualization of small world networks. *IEEE Symposium on Information Visualization, InfoVis 2003*, 75-81, 19-21 Oct. 2003.

[25] Shrinivasan, Yedendra B., van Wijk, Jarke J. VisPad: Integrating Visualization, Navigation and Synthesis *IEEE Symposium on Visual Analytics Science and Technology (VAST 2007)*, 209-210, 2007.

[26] Buono, P., Plaisant, C., Simeone, A., Aris, A., Shneiderman, B., Shmueli, G., Jank, W. Similarity-Based Forecasting with Simultaneous Previews: A River Plot Interface for Time Series Forecasting. *Proc. of the 11th International Conf. on Information Visualisation (IV'07)*, 191-196, Zurich, Switzerland; 2-6 July 2007.