# Interactive Network Exploration to Derive Insights: Filtering, Clustering, Grouping, and Simplification

Ben Shneiderman and Cody Dunne

University of Maryland, College Park, MD, USA
{ben,cdunne}@cs.umd.edu

**Abstract.** The growing importance of network analysis has increased attention on interactive exploration to derive insights and support personal, business, legal, scientific, or national security decisions. Since networks are often complex and cluttered, strategies for effective filtering, clustering, grouping, and simplification are helpful in finding key nodes and links, surprising clusters, important groups, or meaningful patterns. We describe readability metrics and strategies that have been implemented in NodeXL, our free and open source network analysis tool, and show examples from our research. While filtering, clustering, and grouping have been used in many tools, we present several advances on these techniques. We also discuss our recent work on motif simplification, in which common patterns are replaced with compact and meaningful glyphs, thereby improving readability.

**Keywords:** Network visualization, visual analytics, readability metrics, dynamic filters, link clustering, attribute grouping, motif simplification

## 1 Introduction

The Graph Drawing community has an admirable history in developing scalable algorithms that present graphs with appealing aesthetic properties. Early work dealt with synthetic or abstract graphs whose drawings were assessed by balance, symmetry, and visual appeal [5,10,11]. But during the past two decades the community has increasingly expanded its scope to deal with the growing number of realistic networks generated by scientific article citations, cell phone calling patterns, banking transactions, social media communications, etc. (e.g., [14]). These networks represent important scientific, commercial, terrorist, or friendship interactions, where pro-social initiatives, commercial enterprises, or entertainment activities can have profound societal impacts. However, criminals or terrorists may use networks for destructive purposes while malicious actors may undermine effectiveness by disrupting or spamming the network.

The growing number of network analysts must cope with complex structures, disconnected components, well-connected clusters, and multiple attributes for nodes and links. They often deal with networks of dynamically evolving structures, where links act as pathways for volatile information or commodity flows.

This shift from abstract problems that appeal to mathematicians and algorithm developers to pressing problems that influence commerce, politics, or national security has dramatically raised the prominence of graph drawing, network analysis, and interactive information visualization. Interactive approaches have become important, because the scale and richness of information within a network means that drawing one aesthetically pleasing graph is rarely the best strategy. Analysts learn complex exploratory processes that include rapid selection of meaningful subsets, relevant groups, and pertinent clusters. The goals for network visualization designers are now more closely tied to readability and the capacity to quickly extract meaningful insights about the data to support time-critical and sometimes life-critical decisions.

The good news for the Graph Drawing community is that our work is in high demand and that pressing new problems appear daily. However, these demands come with raised expectations, forcing researchers to simultaneously address basic and applied problems, theory and practice, and curiosity-driven as well as mission-driven agendas.

Evaluation methods are also changing. Algorithmic efficiency remains vital, but on actual graphs, machine performance may be improved by leveraging knowledge of the topology. Worst case analyses are still helpful but performance on frequently occurring examples is also important. Another change is that human performance by domain experts and novice analysts to extract insights becomes increasingly important compared with machine performance [21,23,25,30].

There is increased interest in automatable measures that can be used to reduce flaws such as node occlusions, link crossings, and node-link overlaps, as these readability metrics or aesthetic criteria can help analysts to create more comprehensible networks. Readability metrics and interactive tools allow users to automatically or manually clean up cluttered diagrams, while facilitating basic tasks such as counting the nodes, following the links, or identifying meaningful clusters. Other challenging tasks include finding nodes or links with extreme attribute values, surprising link attributes, or unexpectedly strong connections among subgroups.

The single goal of drawing the best graph gives way to developing systematic yet flexible discovery processes [24,25] that support the collection of valuable insights. A starting step is data cleaning to ensure that missing data problems are resolved, data values are within permissible ranges, no duplicate data have crept into the archive, and known constraints across data values are respected. Then systematic yet flexible discovery processes enable users to apply step-by-step methods to explore a network, ensuring that they conduct appropriate tests, while allowing detours when interesting patterns invite further attention.

Modern graph drawing challenges are increasingly complex because of the growing availability and variety of node and link attributes. Sometimes these are numerical quantities that can be conveniently represented by node size, node color, link thickness, or link color. Other times, node or link labels contain meaningful data such as names, text strings, or entire documents, which can sometimes be displayed on the graph or explored one at a time with tooltips.

However, these node and link labels often require integration of increasingly sophisticated natural language processing or machine learning and data mining algorithms. In general, the rich integration of visualization with text analytics and statistics is providing high payoffs for analysts [9,29]. Sometimes the visualization will suggest to users which analytics methods should be applied, other times the analytics methods suggest which visualizations might be most helpful.

We have found ways to help users solve these problems with interactive methods, such as (1) filtering to remove less important nodes and links, (2) clustering by link structures to identify structural components, (3) grouping by node values to understand groups and their relationships, and (4) motif simplification to reduce complexity and increase readability by replacing common structures with simpler glyphs.

To attain the full benefits of these methods, users often need novel layout algorithms and multiple coordinated views. Force-directed layouts are being steadily improved to deal with special cases such as near-planar graphs, disconnected components, or high link-density graphs.

More attractive opportunities may lie in slicing and dicing graphs into meaningful clusters, based on connectivity, or into well-defined groups, based on node attributes. Once clusters and groups have been formed, they can be placed in different windows or panes, so as to permit exploration of each component as well as its relationships with other components.

For several years our group has been embedding our ideas into NodeXL [31] (`www.codeplex.com/nodexl`), a free open-source interface that allows users to draw graphs by using a Microsoft Excel template. We provide a textbook to introduce concepts and guide new users as well as knowledgeable analysts [15]. A large number of example networks collected and analyzed with NodeXL are available on an open data web site (`www.nodexlgraphgallery.org`). For programmers, we provide a set of reusable C# class libraries, a WPF graph visualization control, and an architecture for custom data import plugins.

This paper focuses on recent improvements we made to NodeXL which were designed to produce more readable and understandable graph drawings. These improvements might help analysts make meaningful insights more reliably, so as to support personal, business, legal, scientific, or national security decisions.

## 2 Defining and Measuring Readability

A happy situation occurs when a small planar graph can be drawn that is free from readability-inhibiting flaws and presents clear color and size encodings, plus readable labels. Since many graphs have high link densities with complex structures, we aspire to create imperfect but useful graph drawings with as few flaws as possible. Put positively, we aspire to enable analysts to uncover structural properties, important clusters, and significant nodes and links.

Users of network analysis tools generally have two techniques for improving the node-link drawing readability: automatic layout algorithms and direct manipulation. Both these approaches are based on heuristics, either defined by the layout algorithm or by the user's mental model of readability. Many researchers focus on improved layout algorithms, but few deal with educating users about readability and helping them effectively manipulate the drawing. We can give users rules to apply (e.g., [32], pp. 13), or teach them about basic readability principles [2]. However, it is challenging for users to balance (or even remember) these rules or optimize the ones that would most benefit the target task.

Defining practical readability measures accelerates progress towards improved graph drawings, because it guides analysts in making both manual and automatic changes that improve quality. One approach is to count the number of flaws that inhibit readability [27] and support analysts in reducing their prevalence [2,7,12].

Occluded nodes, crossed links, or node-link overlaps clearly interfere with readability, so measuring their frequency is a good starting point. But even with three metrics analysts have to make difficult tradeoffs when revisions to improve one of these metrics degrades another or modifies the spatial layout substantially. These modifications can have a profound impact on the detection of communities and the perceived importance of individual nodes [20,22]. Hence, significant thought must be given to properly drawing graphs so that analysts will be able to understand and effectively communicate data like clusters, the bridges between them, and the importance of individual nodes. Since more than a dozen graph readability metrics have been defined, analysts would do well to create taxonomies of graphs and tasks so that the weighting of each metric might be tuned to the graph or task type.

We have begun building a task taxonomy of readability metrics and a NodeXL implementation, so that users can calculate metrics useful for the task at hand. In addition to global metrics for the entire drawing, we are implementing local metrics for individual nodes, links, or groups so as to direct users to problem areas that need attention. These local metrics can be updated as users manipulate the drawing, and used to color-code problem areas. Moreover, they can be fed into automatic layout algorithms with multiple criteria [5] or to provide semi-automatic "snap-to-grid" assistance that reduces users' manipulation burden.

## 3 Improving Readability

This paper describes four interactive methods for improving readability and enabling analysts to more frequently create readable graphs that enable them to extract meaningful insights: (1) filtering to remove less important nodes and links, (2) clustering by link structures to identify structural components, (3) grouping by node values to understand groups and their relationships, and (4) motif simplification to reduce complexity and increase readability by replacing common structures with simpler glyphs. Table 1 summarizes the goals these four methods help realize.

### 3.1 Filtering to remove less important nodes and links

Some networks have large numbers of nodes and links which can obscure meaningful groups or network items with interesting attribute values. User-controlled

Table 1: Interactive methods to reveal patterns.

| Method | Entity of attention | Goals |
|---|---|---|
| **Filtering** | Node & link attribute values or statistics | Remove less relevant nodes & links to reveal relevant patterns and key nodes & links |
| **Clustering** | Clusters algorithmically based on link connectivity | Identify structural components, then redraw to highlight clusters; replace clusters with single nodes; show group size and inter-group relationships (color, size, group-in-a-box) |
| **Grouping** | Groups based on node attributes | Use pre-existing groups, then redraw to highlight groups; replace groups with single nodes; show group size and inter-group relationships (color, size, group-in-a-box) |
| **Simplification** | Common, meaningful structures to replace with simplified glyphs | Reduce visualization complexity, raise visibility of common structures, reveal occluded structures, allow comparisons |

dynamic query filters [1,35] have demonstrated their value in successful commercial products that deal with multivariate data, such as Spotfire and Tableau. Dynamic query filters are even more valuable in network visualizations, where the clutter of nodes and links can severely inhibit readability. NodeXL supports filters on node values, link values, graph metrics, layout positions, and many other attributes.

The power of filtering is shown in an example network of U.S. Senate voting patterns from 2007.[1] The similarity in voting patterns (from 0.0 to 1.0) is an attribute of each one of the 4950 links connecting the 100 Senator nodes. The naive drawing produces a thickly connected graph (Fig. 1), but filtering the similarity values to show only those with values above 0.65 produces a revealing portrait (Fig. 2). The force-directed layout further shows the willingness of three Republican Senators (center, in red) to vote in support of their Democrat colleagues (top-right, in blue). One of these, Arlen Specter, later switched his affiliation to the Democrats in 2009.

Filtering can be applied to node values as well to remove incidental nodes, leaving only key actors. Filtering is a well-established technique for multivariate data, as shown in scattergrams, but the variety of filters in many networks means careful thought is needed to produce effective results. Furthermore, scattergram filtering typically leaves the remaining markers in place, but in networks, layout methods interact with filtering, so thoughtful exploration is needed.

[1] Data provided by Chris Wilson of Slate magazine available in the NodeXL template format at http://goo.gl/oa4tg

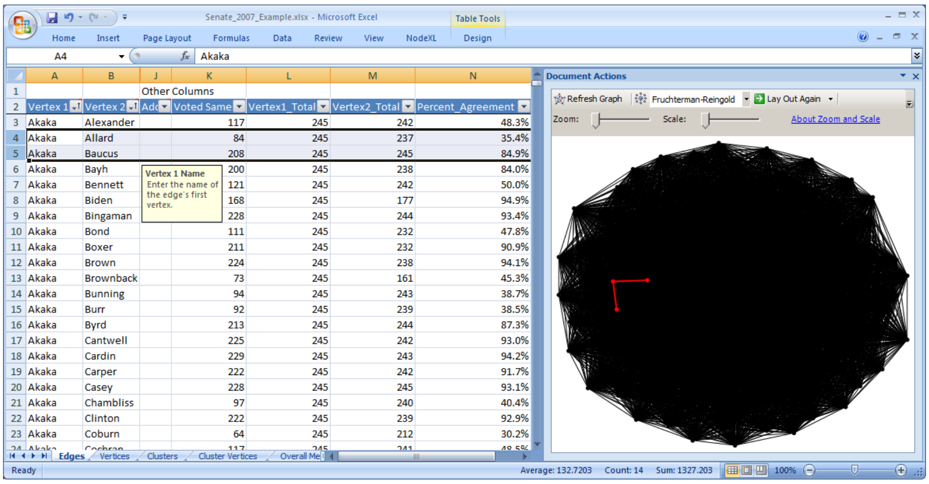



Fig. 1: 2007 U.S. Senate voting graph, showing all 4950 links.

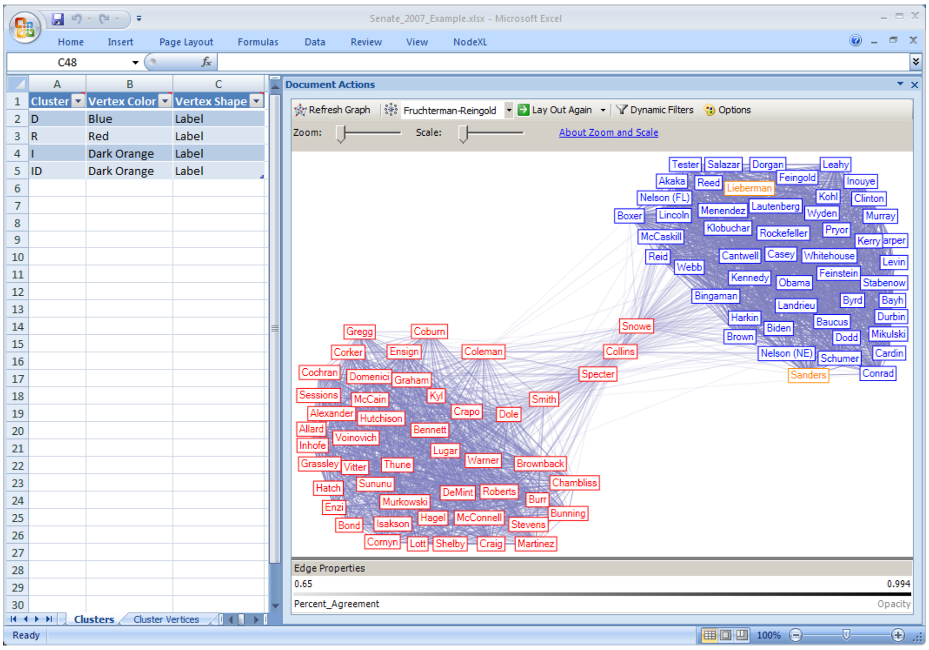



Fig. 2: 2007 U.S. Senate voting graph voting graph, showing only links with a voting similarity value greater than 0.65. Republican Senators Snowe, Collins, and Specter (center, in red) have voting patterns that are closer to their Democrat colleagues (top-right, in blue). Specter ended up switching his affiliation to Democrat in 2009. Independents are in orange and laid out as part of the top-right, blue Democrat community.

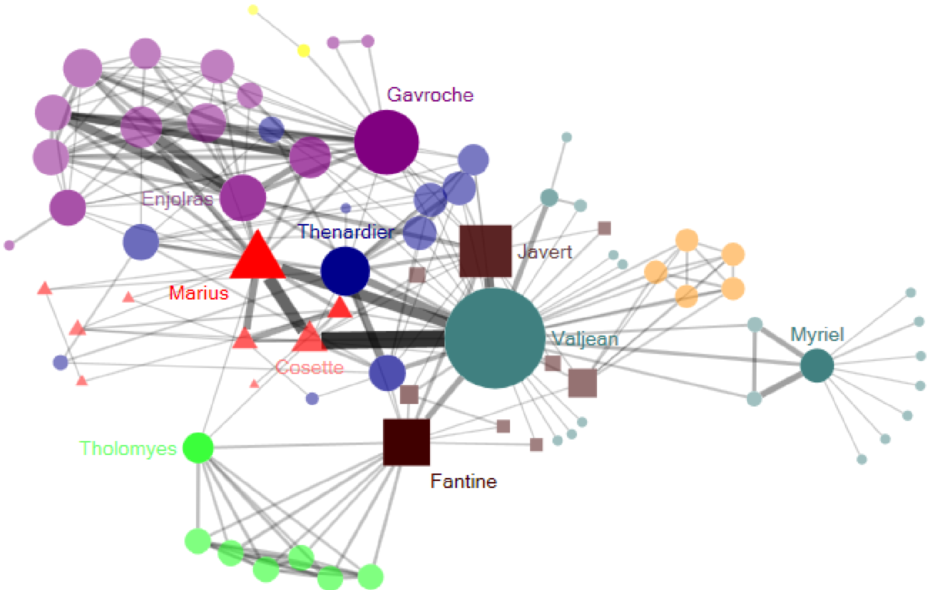



Fig. 3: Co-appearance network in *Les Misérables*.

### 3.2 Clustering to identify structural components

Understanding the complexity of human anatomy is often facilitated by decomposing into subsystems such as circulatory, muscular, skeleton, neural, digestive, etc. These decompositions favor functional structures over physical adjacency. Since networks represent complex phenomena, clustering by link connectivity into functional subsystems often proves to be beneficial.

Clustering by link connectivity has been a popular method, but rapid developments in the algorithms show that much work remains. NodeXL implements the Clauset-Newman-Moore [4], Wakita-Tsurumi [33], and Girvan-Newman [13] clustering algorithms, which all result in mutually exclusive cluster membership. They currently work only on undirected graphs, but additions to support directed and weighted graphs are planned. An example of clustering is the network of characters in *Les Misérables*, shown in Fig. 3. Link strength shows the number of scenes in which pairs of characters appear, while node size shows the number of scenes for each character. The co-appearance network shows the relatedness among characters.

Improvements in clustering algorithms are likely, and their integration with filtering would bring further advances. For example, it might be helpful to show that certain filtering settings would produce dramatically more effective clusters. Verifying the quality of a clustering outcome is often hampered by the lack of a ground truth. Clustering is often used as an exploratory data analysis method to discover unexpected inclusions within a known cluster, unexpected separation into other clusters, or surprising clusters.

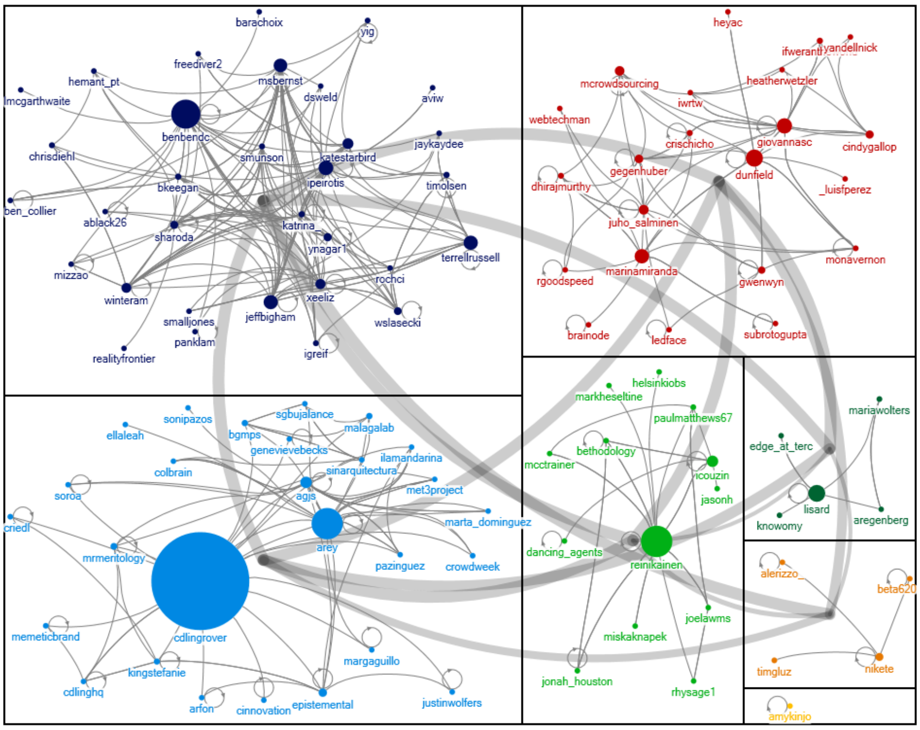



Fig. 4: Twitter event graph for Collective Intelligence 2012.

Unfortunately, layout algorithms may not help analysts to see the clusters clearly. To reinforce the cluster membership, the NodeXL team added the **group-in-a-box** feature [28]. The clusters are placed inside a box whose size depends on the number of nodes, and arranged by the squarified treemap algorithm [3]. While NodeXL does not currently support hierarchical clustering, the treemap layout could be easily extended to visualize nested clusters. Fig. 4 shows the 7 clusters identified for Twitter users who used the hashtag #CI2012 to indicate the Collective Intelligence conference held in April 2012. Links indicate follower, following, mentions, or retweet relationships, and node size indicates number of followers. The largest cluster in the upper left consists mainly of US-based researchers, while other clusters represent other national groups and commercial users. The link strength between clusters is indicated by a "combined" link whose thickness indicates number of links.

Since large numbers of links that span a graph drawing can undermine readability, there has been a strong attraction to link bundling to reduce clutter [19,26]. Fig. 5 shows 12 clusters, in which the inter-cluster links are bundled together. The initial view is attractive, but the bundles seem to obscure rather than highlight the strength of relationships among the clusters. Improvements to link bundling strategies seem possible.

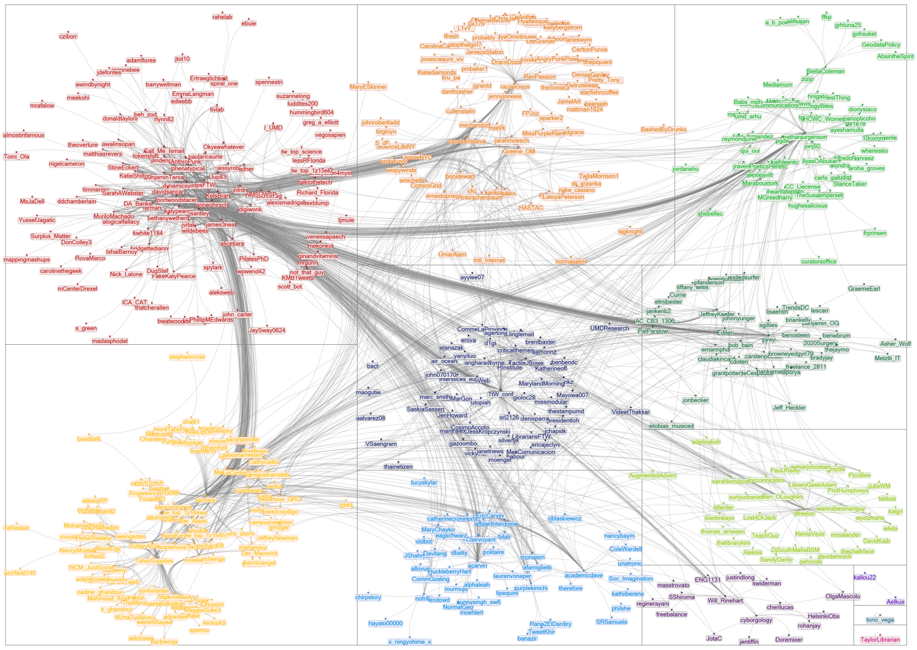


Fig. 5: Twitter event graph for Theorizing the Web 2012 (by Awalin Sopan).

### 3.3 Grouping to find attribute relationships

Nodes may represent people, places, documents, or roles, which are readily understandable in small networks. However, with thousands or millions of nodes, analysts may gain insights by replacing nodes of a common type with a single group node, e.g. author nodes in a scientific citation network might be grouped by their current institution into a single node for each institution. This node could be sized by the number of authors, thereby showing the productive institutions and revealing the degree of collaboration across institutions. Simplifying a million-node author network into a 3000 node institution network removes some information, but reveals important patterns.

Attribute-based node aggregation has been leveraged by several tools to understand overall relationships at the expense of showing the underlying topology explicitly. PivotGraph [34] groups nodes based on the intersection of a pair of attributes, and arranges the meta-node for each group on a grid with each attribute as an axis. Aggregate links between groups are shown with arcs. Similarly, GraphTrail [6] groups nodes by attribute into standard charts, where the groups can be further filtered, merged, or used to pivot to connected groups of other node types. One advantage of this aggregation is a dramatic reduction in screen space required, a fact leveraged by GraphTrail to show the history of exploration directly integrated into the network analysis canvas.

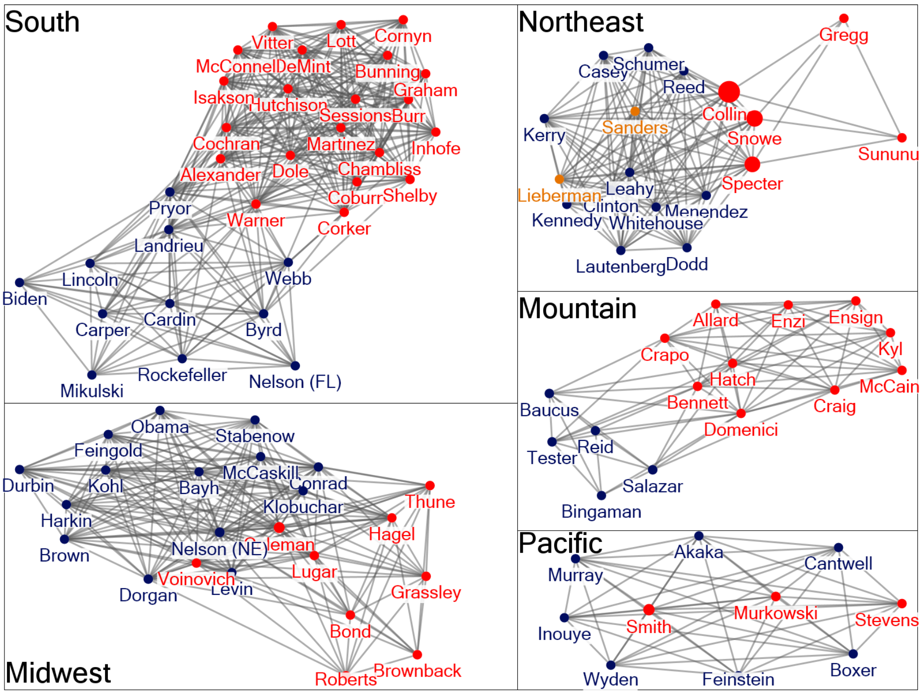



Fig. 6: 2007 U.S. Senators grouped by their regional affiliation (from [28]).

In NodeXL, grouping by node values is often the first step in creating a group-in-a-box layout that preserves individual node visibility, while enabling analysts to see relationships within groups. Fig. 6 shows U.S. Senators (from Fig. 1) grouped by their regional affiliation and presented in a group-in-a-box layout, with links between groups removed to reduce complexity. Nodes are also colored by node values to show party affiliations: red for Republicans, blue for Democrats, and orange for independents. Grouping multiple nodes into a single node or removing links between groups produces measurable improvements in readability by removing less relevant information. Alternatively, when nodes are grouped in a group-in-a-box layout, links can then be combined as in Fig. 4.

### 3.4 Motif simplification to reduce complexity

Many networks have repeated occurrences of familiar **motifs**, which are common patterns of nodes and links. These motifs may be a natural part of the network structure or merely an artifact of the data collection process. Regardless of their cause, many frequently expressed motifs contain little information, especially when compared to the space they occupy in the network visualization. Rather than asking users to view the entire network or to filter out specific subgraphs manually, NodeXL provides tools to automatically identify several common motifs and simplify them into representative glyphs [8]. Well-designed glyphs have several benefits: they (1) require less screen space, (2) are easier to understand in the context of the network, (3) can reveal otherwise hidden relationships, and (4) preserve as much underlying information as possible.

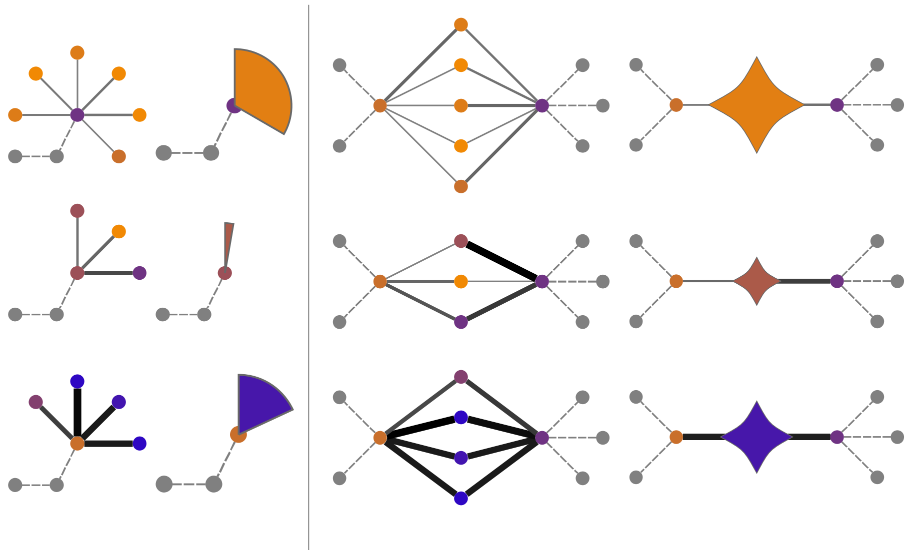
Our initial work focused on three basic motifs that are especially valuable to social scientists:

- A **fan motif**, sometimes called a star, consists of a **head node** connected to **leaf nodes** that have no other neighbors (Fig. 7a). Since there may be hundreds of leaves, replacing all the leaves and their links to the head with a simple **fan glyph** can dramatically simplify a network.
- A **connector motif** consists of a set of functionally equivalent **span nodes** that solely span two or more **anchor nodes** (Fig. 7b). Simplifying the connector motifs frequently reduces the complexity of the dense center of the node-link diagram, and allows easier connectivity comparisons.
- A **clique motif** consists of a set of nodes in which each pair is connected by at least one link (not shown). Cliques with four or more nodes are common in dense social networks. Replacing them with representative glyphs makes it easier to understand overall connectivity.

An example of motif simplification is shown in Fig. 8, which represents the bipartite network for the Lostpedia wiki community collected by Beth Foss. Boxes with labels show wiki pages, linked to the colored discs representing their associated editors. The editors are colored and sized according to two measures of their activity in the wiki. The left side of Fig. 8 shows the initial network, while the right shows a simplified version. Each fan motif was replaced with a fan glyph that shows the underlying topology, while each connector motif with two anchors is replaced with a connector glyph that shows the bridging relationship. The simplified view on the right has 25 nodes instead of the original 513, and requires a quarter of the original screen space. Improved readability would come by replacing the connector motifs that have three or four anchors as well. The improvement is even more pronounced in larger datasets with thousands of nodes in a fan and large connector groups.

For each motif we want to simplify, great care must be given to the design of a glyph to represent it. Arbitrary motifs can be shown as a simple meta-node (⨁), but a representative glyph that reveals the motif properties will be far more effective. Some examples of fan and connector motifs and the representative glyphs we designed to replace them are shown in Fig. 7. The shape of our glyphs indicates the underlying topology, and each glyph is sized proportional to the number of nodes it replaces. The arc (and thus the area) of each fan glyph is scaled by the number of nodes it contains, in a range of 10–120°. The size of a connector glyph is also scaled linearly by the number of nodes it replaces. If an underlying visual attribute encoding exists for the nodes or links, we try to show the average of the underlying attribute values using the same color or size scale in the glyph version.

Using motif glyphs we can effectively simplify a static drawing, but interactivity is required for the network to be easily understandable and investigable.



(a) Fan motifs and glyphs

(b) Connector motifs and glyphs

Fig. 7: Three fan motifs (7a) and three connector motifs (7b) are shown here along with their simplified glyph versions. The motifs vary in the number of nodes they contain and the node attribute values used for the blue-orange color scale.

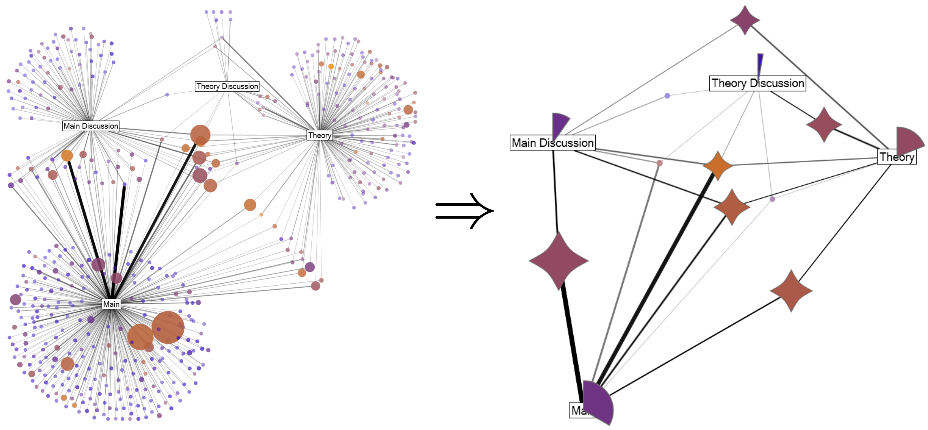



Fig. 8: Edit history of the Lostpedia wiki pages on the "Four-toed-statue" (left) and a simplified drawing with fan and 2-connector motif glyphs (right, zoomed in). Wiki pages are shown as boxes with labels and are connected by the contributors editing them, with some contributors editing only one page and other users editing two or more. Contributors are colored and sized by their total number of edits on main or theory pages. Data by Beth Foss.

In NodeXL, users can investigate the contents of any glyph, switch between the original and simplified version of any set of motifs on demand, position the glyphs manually, and even use automatic layout algorithms on the simplified drawing. Effective layout algorithms for these simplified networks require special handling of non-uniform node sizes and shapes [17].

## 4 Case Study: Finding Regional Innovation Clusters

One of the goals of urban planners is to understand the relationships behind innovation and how the ties between organizations, individuals, and funding agencies affect growth. Christopher Scott Dempwolf,[2] a researcher in the School of Architecture, Planning and Preservation at the University of Maryland, has been working to model innovation based on patent ties, federal and state funding, and physical locations. We introduced Dempwolf to NodeXL and helped guide several of his network analyses, including one of Pennsylvania innovations in 1990. He was keen on detecting technology and talent clusters, which could then be positively influenced. The network he collected included patent ties, federal funding from SBIR/STTR, and state funding through the DCED and Ben Franklin Technology Partners.

An initial drawing of this network is shown in Fig. 9, which uses the Harel-Koren layout [16,18], link bundling, and categorical coloring for node and link types. While quite beautiful, this drawing is not particularly effective. Some large structures are easily distinguishable, like the cauliflower-shaped groups of gray inventors and a few large orange enterprises. However, the overall structures and relationships are hard to interpret.

Dempwolf was interested in technology and talent clusters, so to try to pick these features out of this large network we applied the Clauset-Newman-Moore clustering algorithm [4]. The algorithm finds clusters of nodes that link to each other more frequently than outside the cluster, which, in this case, represents clusters of entities with similarities in patented technology. With a node-link diagram alone it can be challenging to see group membership, size, and aggregate relationships using solely color or shape coding. We applied the group-in-a-box layout [28] to make these features explicitly visible by laying out each detected cluster individually, after filtering out several tiny clusters (Fig. 10).

In analyzing this drawing, we discovered many expected clusters around specific Pennsylvania counties and local enterprises. For example, the bottom-left cluster of Fig. 10 is the Pittsburgh metro area, containing the large orange Westinghouse Electric. The Pittsburgh cluster is highly connected (via the hidden links) to the Montgomery county cluster to its right, another large metro area. An unexpected exception to the location grouping is the top-left pharmaceutical and medical cluster, composed of several companies, universities, HHS, and an interesting arrangement of inventors in several connected fans. These sorts of meaningful structures were mostly hidden in the original drawing (Fig. 9).

[2] http://www.terpconnect.umd.edu/~dempy

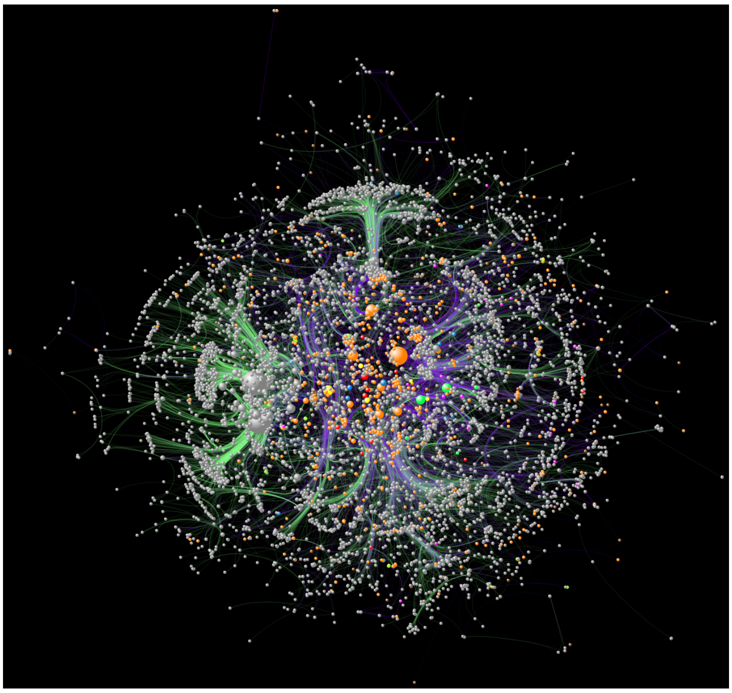


Fig. 9: Pennsylvania innovation relationships in 1990 (data by Scott Dempwolf).

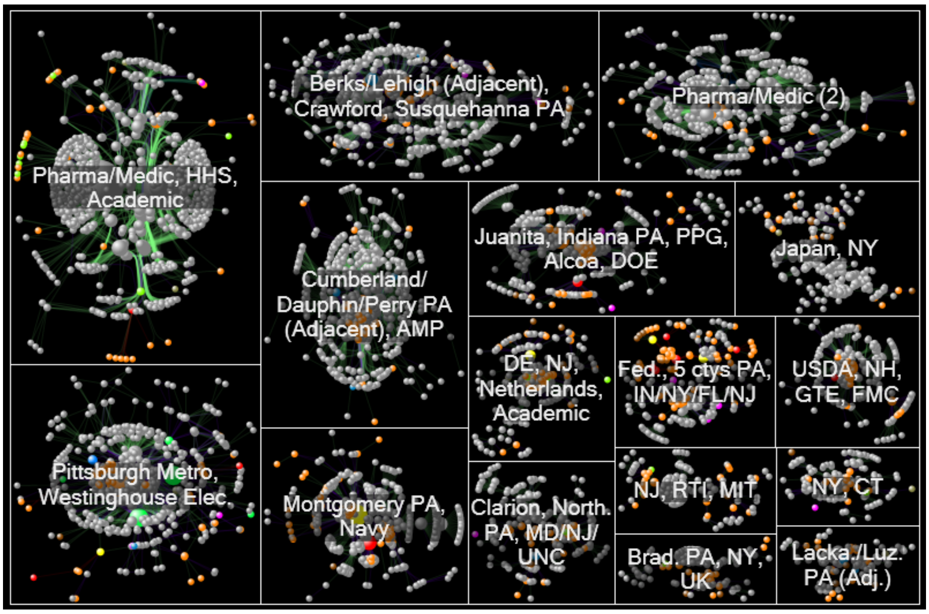



Fig. 10: The innovation network from Fig. 9 after using the group-in-a-box layout, hiding inter-group links, and filtering to only the largest groups.

Dempwolf found that these clusters represented specific economic development opportunities that could be influenced to increase employment. According to him, "This approach gives you a list of firms to go talk to and specific things to talk with them about. It also identifies specific talent clusters. These are things that traditional industry cluster analysis has never done." More details of Dempwolf's use of NodeXL for identifying high-priority economic development targets are available in his slide deck.[3]

## 5 Discussion

Interactive network analysis is of growing importance for many national priorities. However, existing visualization tools often show cluttered, dense drawings from which analysts have difficulty deriving insights. Making it easier for a wide range of users to succeed with network analysis would dramatically expand its application across diverse disciplines. Our tool NodeXL is already widely used, especially for introductory courses, but improving the readability of network drawings would greatly increase its value.

This paper demonstrates how classic techniques of dynamic filtering, link clustering, and attribute-based node grouping can be effectively integrated in NodeXL. Moreover, our new motif simplification technique enables rapid, interactive complexity reduction, so as to more clearly present the underlying network structure. NodeXL also includes readability metrics users can use to gauge the effectiveness of their drawings, or even feed into automated techniques to improve the layout. Moreover, the metrics can be used to quantify the benefit of our filtering, clustering, grouping, and simplification techniques.

**Acknowledgments.** The authors wish to thank Marc Smith and the NodeXL team for their support. This project is supported by National Science Foundation grants 0915645 and 0968521, the Social Media Research Foundation, and the Connected Action Consulting Group.

## References

1. Ahlberg, C., Williamson, C., Shneiderman, B.: Dynamic queries for information exploration: an implementation and evaluation. In: CHI '92: Proc. SIGCHI conference on Human Factors in Computing Systems. pp. 619–626 (1992), DOI:10.1145/142750.143054
2. Bonsignore, E.M., Dunne, C., Rotman, D., Smith, M., Capone, T., Hansen, D.L., Shneiderman, B.: First steps to NetViz Nirvana: Evaluating social network analysis with NodeXL. In: CSE '09: Proc. 2009 international conference on computational science and engineering. vol. 4, pp. 332–339 (2009), DOI:10.1109/CSE.2009.120
3. Bruls, M., Huizing, K., J. Van Wijk, J.: Squarified Treemaps. In: Proc. Joint Eurographics and IEEE TCVG symposium on Visualization. pp. 33–42 (2000)

[3] http://portal.sliderocket.com/ATWBE/Using-SNA-to-find-and-manage-RICs

4. Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Physical Review E: Statistical, Nonlinear, and Soft Matter Physics 70, 066111 (2004), DOI:10.1103/PhysRevE.70.066111
5. Davidson, R., Harel, D.: Drawing graphs nicely using simulated annealing. TOG: ACM Transactions on Graphics 15(4), 301–331 (1996), DOI:10.1145/234535.234538
6. Dunne, C., Riche, N.H., Lee, B., Metoyer, R.A., Robertson, G.G.: GraphTrail: Analyzing large multivariate, heterogeneous networks while supporting exploration history. In: CHI '12: Proc. 2012 international conference on Human Factors in Computing Systems. pp. 1663–1672 (2012), DOI:10.1145/2207676.2208293
7. Dunne, C., Shneiderman, B.: Improving graph drawing readability by incorporating readability metrics: A software tool for network analysts. Human-Computer Interaction Lab Tech Report HCIL-2009-13, University of Maryland (2009)
8. Dunne, C., Shneiderman, B.: Motif simplification: Improving network visualization readability with fan and parallel glyphs. Human-Computer Interaction Lab Tech Report HCIL-2012-11, University of Maryland (2012)
9. Dunne, C., Shneiderman, B., Gove, R., Klavans, J., Dorr, B.: Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization. JASIST: Journal of the American Society for Information Science and Technology (2012)
10. Eades, P.: A heuristic for graph drawing. CN: Congressus Numerantium 42, 149–160 (1984)
11. Fruchterman, T.M.J., Reingold, E.M.: Graph drawing by force-directed placement. SPE: Software: Practice and Experience 21(11), 1129–1164 (1991), DOI:10.1002/spe.4380211102
12. Gansner, E., Hu, Y.: Efficient node overlap removal using a proximity stress model. In: GD '08: Proc. 16th International Symposium on Graph Drawing. pp. 206–217 (2009), DOI:10.1007/978-3-642-00219-9_20
13. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PNAS: Proc. National Academy of Sciences of the United States of America 99(12), 7821–7826 (2002), DOI:10.1073/pnas.122653799
14. Hachul, S., Jünger, M.: An experimental comparison of fast algorithms for drawing general large graphs. In: GD '05: Proc. 13th International Symposium on Graph Drawing. vol. 3843/2006, pp. 235–250 (2006), DOI:10.1007/11618058_22
15. Hansen, D., Shneiderman, B., Smith, M.: Analyzing social media networks with NodeXL: Insights from a connected world. Morgan Kaufmann (2011)
16. Harel, D., Koren, Y.: A fast multi-scale method for drawing large graphs. JGAA: Journal of Graph Algorithms and Applications 6(3), 179–202 (2002)
17. Harel, D., Koren, Y.: Drawing graphs with non-uniform vertices. In: AVI '02: Proc. Working Conference on Advanced Visual Interfaces. pp. 157–166 (2002), DOI:10.1145/1556262.1556288
18. Harel, D., Koren, Y.: Graph drawing by high-dimensional embedding. In: GD '02: Proc. 10th International Symposium on Graph Drawing. pp. 299–345 (2002), DOI:10.1007/3-540-36151-0_20
19. Holten, D.: Hierarchical edge bundles: visualization of adjacency relations in hierarchical data. TVCG: IEEE Transactions on Visualization and Computer Graphics 12(5), 741–748 (2006), DOI:10.1109/TVCG.2006.147
20. Huang, W., Hong, S.H., Eades, P.: Layout effects on sociogram perception. In: GD '05: Proc. 13th International Symposium on Graph Drawing. vol. 3843/2006, pp. 262–273 (2006), DOI:10.1007/11618058_24

21. Lam, H., Bertini, E., Isenberg, P., Plaisant, C., Carpendale, S.: Empirical studies in information visualization: Seven scenarios. TVCG: IEEE Transactions on Visualization and Computer Graphics PP(99), 1 (2011), DOI:10.1109/TVCG.2011.279
22. McGrath, C., Blythe, J., Krackhardt, D.: The effect of spatial arrangement on judgments and errors in interpreting graphs. SN: Social Networks 19(3), 223–242 (1997), DOI:10.1016/S0378-8733(96)00299-7
23. North, C.: Toward measuring visualization insight. CGA: IEEE Computer Graphics and Applications 26(3), 6–9 (2006), DOI:10.1109/MCG.2006.70
24. Perer, A., Shneiderman, B.: Systematic yet flexible discovery: Guiding domain experts through exploratory data analysis. In: IUI '08: Proc. 13th International Conference on Intelligent User Interfaces. pp. 109–118 (2008), DOI:10.1145/1378773.1378788
25. Perer, A., Shneiderman, B.: Integrating statistics and visualization for exploratory power: From long-term case studies to design guidelines. CGA: IEEE Computer Graphics and Applications 29(3), 39–51 (2009), DOI:10.1109/MCG.2009.44
26. Pupyrev, S., Nachmanson, L., Bereg, S., Holroyd, A.E.: Edge routing with ordered bundles. In: GD '11: Proc. 19th International Symposium on Graph Drawing. pp. 136–147 (2011), DOI:10.1007/978-3-642-25878-7_14
27. Purchase, H.C.: Metrics for graph drawing aesthetics. JVLC: Journal of Visual Languages & Computing 13, 501–516 (2002), DOI:10.1006/jvlc.2002.0232
28. Rodrigues, E.M., Milic-Frayling, N., Smith, M., Shneiderman, B., Hansen, D.: Group-in-a-Box layout for multi-faceted analysis of communities. In: SocialCom '11: Proc. 2011 IEEE 3rd International Conference on Social Computing. pp. 354–361 (2011), DOI:10.1109/PASSAT/SocialCom.2011.139
29. Shneiderman, B.: Inventing discovery tools: combining information visualization with data mining. In: DS '01: Proc. 4th International Conference on Discovery Science. pp. 17–28 (2001), DOI:10.1007/3-540-45650-3_4
30. Shneiderman, B., Plaisant, C.: Strategies for evaluating information visualization tools: Multi-dimensional in-depth long-term case studies. In: BELIV '06: Proc. 2006 AVI workshop on BEyond time and errors: novel evaLuation methods for Information Visualization. pp. 1–7 (2006), DOI:10.1145/1168149.1168158
31. Smith, M., Shneiderman, B., Milic-Frayling, N., Rodrigues, E.M., Barash, V., Dunne, C., Capone, T., Perer, A., Gleave, E.: Analyzing (social media) networks with NodeXL. In: C&T '09: Proc. fourth international conference on Communities and Technologies. pp. 255–264 (2009), DOI:10.1145/1556460.1556497
32. Sugiyama, K.: Graph drawing and applications for software and knowledge engineers, vol. 11. World Scientific Publishing Company (2002)
33. Wakita, K., Tsurumi, T.: Finding community structure in mega-scale social networks: [extended abstract]. In: WWW '07: Proc. 16th international conference on World Wide Web. pp. 1275–1276 (2007), DOI:10.1145/1242572.1242805
34. Wattenberg, M.: Visual exploration of multivariate graphs. In: CHI '06: Proc. SIGCHI conference on Human Factors in Computing Systems. pp. 811–819 (2006), DOI:10.1145/1124772.1124891
35. Williamson, C., Shneiderman, B.: The dynamic HomeFinder: evaluating dynamic queries in a real-estate information exploration system. In: SIGIR '92: Proc. 15th annual international ACM SIGIR conference on research and development in information retrieval. pp. 338–346 (1992), DOI:10.1145/133160.133216