Improving Graph Visualization in TechNotes' GraphView

for
Braden MacDonald
CEO
TechNotes
Vancouver, British Columbia

By
Konstantin Mestnikov
ENGL301 Technical Writing Student
E-mail: konstantin@student.ubc.ca

July 20, 2022

Table of Contents

Figures and Tables

Abstract

TechNote's Graph View allows for the database entries to be represented as a graph. However, problems with cluttering can make data navigation and exploration difficult. To address this problem, several graph decluttering techniques are analyzed, including using an alternative node layout, an alternative edge representation as well as more specialized techniques that involve node type hiding, community detection and clique detection. A survey to assess levels of clutter, perceived difficulty as well as factual understanding of graph data was used to gather data from participants. Based on the results of the survey and cost analysis, the following recommendations are devised for TechNotes:

- Implementing the alternative graph is a single most impactful and cost-effective measure that can improve performance and perception of GraphView
- Making community-based layout a more prominent or default feature when showing entities of a single type can provide better user experience
- Improving clique-based layout by providing more intuitive documentation as to how to use it and making it an option only when the users need to ask specific questions about group clusters would make the best use of this complex layout

Introduction

Background on TechNotes Database and Graph View Issues

The Graph View in TechNotes allows for the data to be represented as a graph, where nodes and edges that stand for entities and relationships in the database. While potentially beneficial to browsing information efficiently and deriving new insights from the data, graphs can quickly become cluttered and hard to interpret.

Visualizations of graphs can become complex and confusing especially when the graph connects entities of different types (companies, concepts, products, etc.) as is the case in TechNotes. In addition, without properly handling the case when the number of edges becomes too large, proper database navigation and data exploration in the graph view become hard or even impossible.

Purpose of this Report

This report aims at evaluating various possibilities for graph visualization to reduce graph cluttering in TechNotes' Graph View, analyzing the feasibility of implementation of chosen graph decluttering features, and at proposing recommendations for Graph View's product development based on the findings.

Method of Inquiry

Potential graph decluttering techniques to be tested in this report were designed for the purposes of this study or identified vie secondary sources. A total of 8 members of the general population responded to a survey designed to analyze how various ways of graph layout and design affect what the graphs convey in terms of aesthetics, information, and comprehension. Additionally, monetary and time expenditures for implementing the graph decluttering techniques were estimated to assess feasibility and priorities of implementing the various techniques.

Scope of the Inquiry

The inquiry will cover the following questions:

- How well does the currently used graph visualization conveys information and how is the graph visualization perceived?
- Which graph decluttering techniques would be most effective based on participants' views?
- What are potential costs for implementing the various graph decluttering techniques?

Data Section

Several strategies to achieve graph decluttering are tested in this report, namely: 1) altered graph layout, 2) altered representation of edges, 3) higher-order graph modifications. The first set of strategies simply involve altering the layout of the graph as it is currently presented in Graph View of TechNotes. The altered layout is evaluated against the original layout. The second type of strategies targets the way edges are represented in the graph. Particularly, a type of edge bundling developed by Bach et al. and called confluence drawing is assessed. The last set of strategies involve higher-order graph modifications that are aimed at highlighting specific dimensions of a graph. Sorting nodes into communities algorithmically according to graph topology, highlighting cliques (tight interconnected clusters) or allowing to hide nodes of specific type are all assessed in an example use case of visualizing a research network.

Alternative Graph Layout Strategy

Current graph layout is computed using a standard forced layout algorithm. The forced algorithm does not consider the different node or relationship types that exist in TechNotes data, and thus produces a layout purely out of surface topology of the graph. This, for example, leads to node arrangements, in which TechConcepts hierarchy is not shown explicitly, or in which products, TechConcepts and companies are not arranged in any meaningful way relative to each other. To address these issues, an alternative layout is devised and contains the following features: 1) TechConcept hierarchy tree is put in a central region of the graph, 2) products are arranged on either side of the TechConcept tree, 3) manufacturers are arranged on both peripheries of the

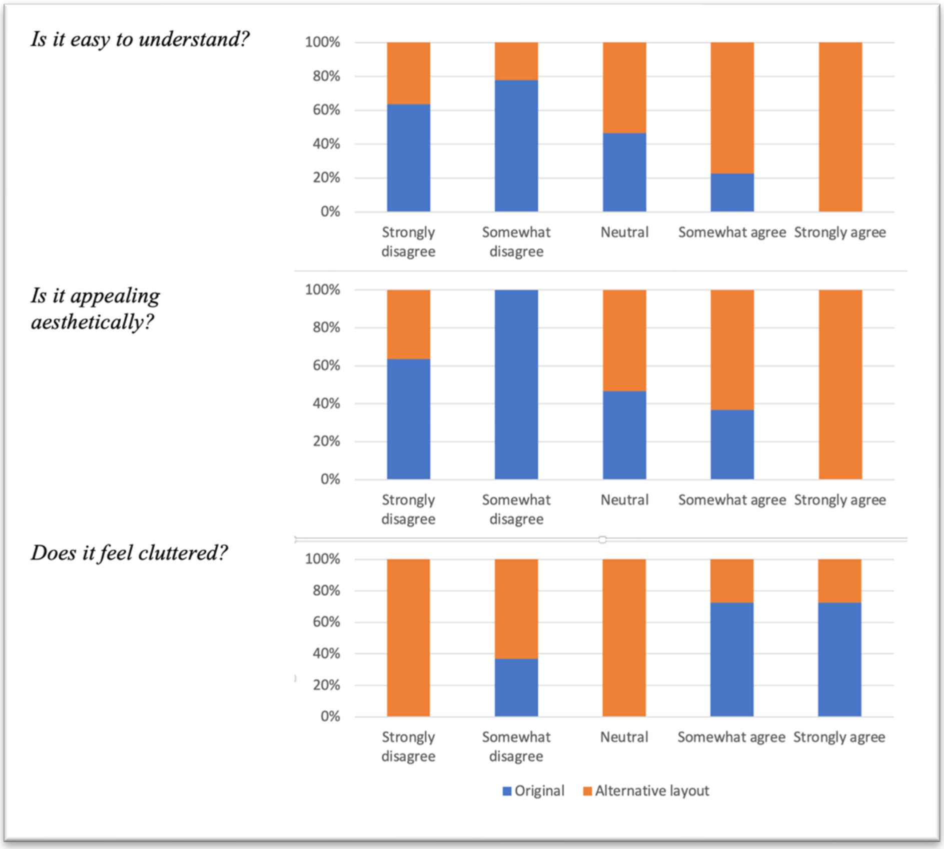

Figure 1 Participants Reply Distributions for Perception-Related Questions

The 100% stacked column chart where orange bar lines represent proportions for the original graph and orange representing the alternative layout results.

graph canvas. To test how this alternative layout compares to the original forced layout, several survey questions were designed that are aimed at assessing participants' perception of aesthetic, complexity, and information factors.

Survey Results. As graph 1 shows, the graph layout is generally perceived as more aesthetically pleasing, easy to understand and less cluttered as the original layout. A further question was aimed at graph comprehension. The participants were asked to name the TechConcept that has the greatest number of products that belong to its type. While using the original layout 86% of participants who answered the question responded "Lithium-ion cell", 40% of participants responded the correct answer "Battery Energy Storage" when prompted with the alternative layout.

Table 1 Percent Correct answers for the Questions Checking Graph Data Comprehension

Highest numerical value in each row is bolded green.

	Original layout, % correct answers	Alternative layout, % correct answers
Is it true that every Battery Energy Storage type product also has Lithiom-ion cells as its part?	38	**57**
Which company produces the most number of products?	63	**71**
Which TechConcept (light blue) has the most number of products that fall under its type?	**86**	40

Discussion of Results. In a striking result, a simple rearrangement of nodes produced a graph that performed better in all perception characteristics that were tested – aesthetics, cluttering and comprehensibility. Other than showing a clear advantage of the alternative graph layout, it shows that perhaps aesthetics and comprehensibility have an underlying relationship that can be further explored in future studies. More than that, alternating the layout allowed for an actual improvement of performance in terms of comprehension, as tested by the "Battery Energy Storage" question. That happened perhaps because arranging various TechNotes entities in a consistent way improved the big picture view and made it easier to distinguish various relationship types as well as amount of a particular entity type.

Confluence Drawings

A different approach to graph decluttering tackles the way edges are represented in a graph. Since the maximum number of edges in a simple graph is proportional to the square of the

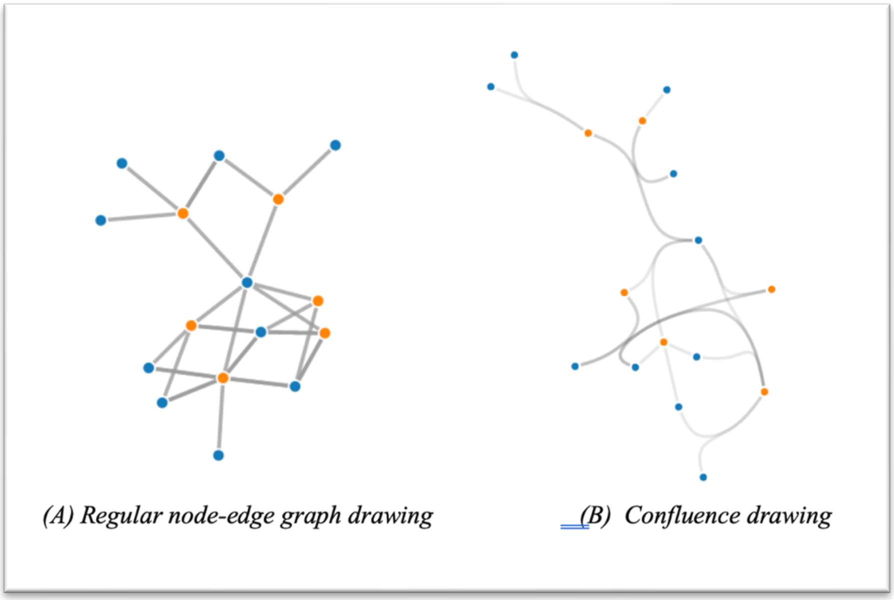

Figure 2 Comparison of Regular and Confluence drawing graph representations

Source: Adapted from Bach et al. 2016

number of nodes, high edge density and high number of edge crossing are common problems that can occur even in relatively small graphs. Various edge bundling techniques are developed to group edges together to reduce redundancy, edge crossing and overall cluttering. In this study, a relatively new way of edge bundling called confluence drawing developed by Bach et al. are assessed. In confluence drawings, edges are processed with an algorithm that bundles the edges and represents them as smooth curves in such a way that if any two edges are connected by one of such smooth curves, it means that they have an edge in between. This leads to a representation where a single curved edge may represent multiple edges at the same time (Figure 2).

Survey Results. Confluence drawing was tested for comprehension by participants via two survey questions. In the first question, participants are presented with a simple node-edge graph without edge bundling and are asked a question. In the second question, participants are presented with a confluence drawing of the identical graph and are asked the same question. Seventy one percent of participants answered the question correctly when presented with a regular graph, while only 28% answered correctly when presented with the confluence drawing version of the graph. Unfortunately, a mistake was found in the second question, so it is disregarded here.

Discussion of Results. Despite the appealing concept of confluence drawings, they fail to provide a less cluttered version of the graph while maintaining comprehensibility. The participants performed much better when presented with a regular graph, than confluence drawing graph, meaning perhaps that confluence drawings are harder to understand and/or take to get accustomed to. This can be explained by the fact that confluence drawing were originally developed for purposes of network science to represent larger graphs, where individual connections are important but not as important as overall structure of the network.

Type Hiding, Cliques, and Communities

The last set of decluttering techniques relate to graph modifications that are specific to graphs that represent networks of individuals, companies, and organizations. In this case, one is interested in asking questions about a network that refers to one type of entities as opposed to many. For example, it could be a network of financial organizations or researchers. In some cases a graph can consist of several entities (such as researchers and their papers), but the focus of the visualization lies in the interactions of one of the entities (for example, how researchers are collaborating).

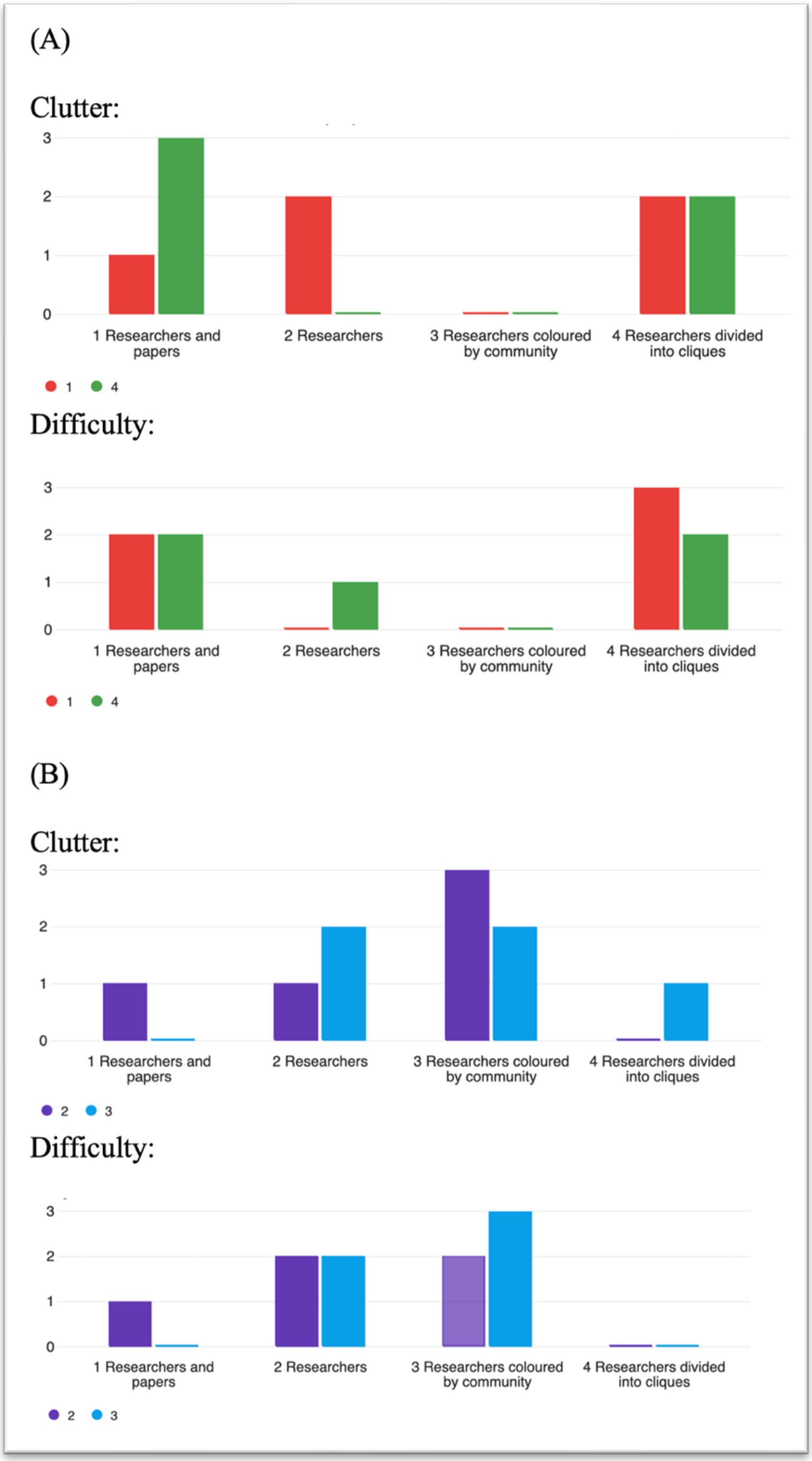

Figure 3 Ranking Results for Difficulty and Clutter Perception

Data summarizes participants' responses for ranking the four graph presentaiton alternatives. For clutter ranking, 1 is most cluttered. For difficulty ranking, 1 is least difficult. Data shows counts for received for the rank indicated by the color. (A) Distributions of ranking for most and least difficult/cluttered graph. (B) Distributions of ranking for middle ranking options (2 and 3).

The techniques used in this section have already been implemented in the system, but their relative effectiveness has not been assessed. The first technique refers to simply hiding every other type of node other than the one under consideration and thus is called *node hiding*. This technique can aid in focusing a particular node type and eliminate all other information altogether.

The second technique combined node hiding with an algorithmic way to automatically detect communities within a given network. A community in this case is something like a cluster and is determined by connectedness of the network. This technique is potentially beneficial as it gives a visual aid in the form of graph nodes coloured by communities for a quick grasp of the topology of the network. The last techniques involve detecting tight node clusters called cliques in addition to detecting communities and node hiding. Overall, the three that will be discussed in this analysis are like Russian dolls, since the clique detection involves community detection, and community detection involves node hiding.

Survey Results. Survey questions were designed to assess perceived difficulty and clutter level of the four graph options. The example graph used in the survey is a network of researchers and papers connected by authorship relationship.

The four graph options for the participants were presented with are:

1. the original graph (“Researchers and papers”)
2. the graph where paper nodes are hidden and only researchers are shown (“Researchers”)
3. the graph where in addition to hiding papers, researchers are broken into communities (“Researchers coloured by community”)

4. the graph with hidden papers, community colouring and detection of cliques ("Researchers divided into cliques")

The participants were asked to rank the four graph options according to perceived difficulty and clutter level. The survey results are presented in Figure 3. To present the discovered patterns better, the rankings are shown separately for the top/bottom (1,4) and middle ranks (2,3). As can be seen in Figure 3 (A), the original layout and the clique layout are equally divided on both sides of the difficult/easy, cluttered/non-cluttered spectra. At the same time, the node-hiding and community versions of layout were mostly in the middle of the spectra for both parameters.

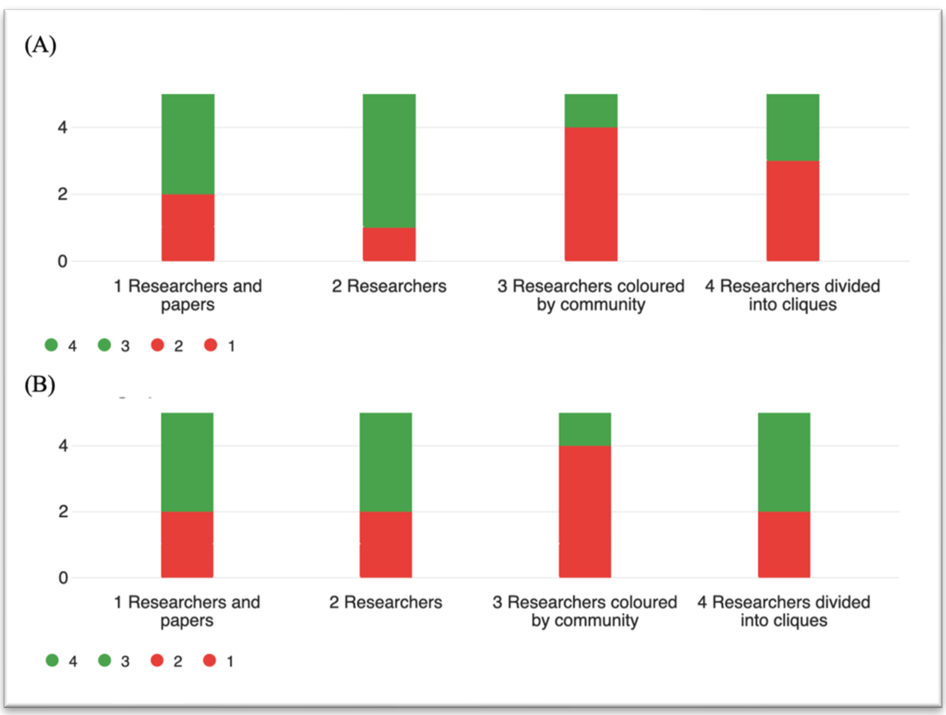

Figure 4 ***Grouped*** Ranking Results for Utility Questions

Data summarizes participants' responses for ranking the four graph presentaiton alternatives. For clutter ranking, 1 is most cluttered. Data shows counts for received for the rank indicated by the color. For difficulty ranking, 1 is least difficult. Top two ranking options are grouped as red, the bottom two options are grouped as green. (A) Question 1 (B) Question 2

Furthermore, the other two questions of the survey aimed at assessing how the different graph layouts are perceived on the scale of usefulness in answering questions about finding different connections in the network. The clique and community detection both outperformed the original and node-hiding layout by number of first and second rank received for the first questions (Figure 4 (A)). For the second question, community detection outperformed the other three methods (Figure 4 (B)).

Discussion of Results. It is interesting to see how original graph and clique detection graph both occur as most difficult and easy/cluttered and non-cluttered according to the survey results. It is possible that the level to which an individual perceives or understands the underlying graph determines their preference. In other words, a more complex layout (clique-based layout) may be seen as easy and neat only if one understands it well. So even a little gap in understanding the layout could significantly decrease the ease of perception. In this case, perhaps the middle options (node-hide and community-based) layouts can be fit for a wider range of audiences, as they avoid "all or none" problem with the clique layout.

Since community and clique-based layouts also performed better in terms of perceived usefulness as seen in Figure 4, the community layout appears as a winner in terms of perceived difficulty, clutter, and usefulness. Its complexity is just right to be comprehensible and appealing and just right to be useful to answer network-based questions.

Cost Analysis of the Tested Decluttering Techniques

Since type-hiding, community detection and finding cliques are already implemented in the system, the cost of employing them is essentially zero. On the other hand, developing the alternative layout described in part a of the Data Section may take considerable time. That is because the layout may need to be adapted to varying and complex data types found in TechNotes and consideration of importance, space, edge crossing etc. will be important in devising the corresponding algorithm. Thus, the layout may take approximately 1-1.5 month of work for a single developer.

The implementation of confluence drawing may be more challenging for several reasons. First, the system on which GraphView is build may not allow such algorithm to be incorporated easily, and in such a case, the entire GraphView may need to be re-implemented using a more flexible framework. Secondly, the algorithm may be difficult to optimize for large-node networks and given that TechNotes database values fast response-speed, it may be challenging and time-consuming to achieve. It may take anywhere from one month, if no changes to the underlying system are necessary, anywhere to three months, if a new framework needs to be used and optimizations for performance made.

Conclusion

Summary and Interpretation of Findings

Several decluttering techniques were examined in this study. The alternative graph layout provides increased performance in graph comprehension as well as improved perception in terms of aesthetics, clutter level and comprehensibility. The confluence drawing approach, while having a good conceptual appeal, suffers from decreased comprehensibility. Cost analysis for implementation the alternative layout shows feasibility of building it in 1-1.5 months, while confluence drawing layout may not be feasible to achieve.

Among the tools for network analysis, the layout that includes community detection with node hiding creates a balanced layout that was found to have higher perceived usefulness for answering network-related questions. On the other hand, the clique-based layout may appear as too complex creating an all-or-nothing response among participants in terms of perceived difficulty, clutter and usefullness. Node-hiding only layout shows a marginal increase in performance compared to unaltered graph layout.

Recommendations for TechNotes' Graph View

Based on the findings as well as cost analysis, the following recommendations for GraphView improvement are made:

- Implementing the alternative graph is a single most impactful and cost-effective measure that can improve performance and perception of GraphView
- Making community-based layout a more prominent or default feature when showing entities of a single type can provide better user experience

- Improving clique-based layout by providing more intuitive documentation as to how to use it and making it an option only when the users need to ask specific questions about group clusters would make the best use of this complex layout

References

Bach, Benjamin, et al. “Towards Unambiguous Edge Bundling: Investigating Confluent Drawings for Network Visualization.” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, 2017, pp. 541–50. *Crossref*, https://doi.org/10.1109/tvcg.2016.2598958.