Visualizing Uncertainty in Graphs

Nathaniel Cesario*
University of California Santa Cruz

ABSTRACT

One problem with visualization is that it can be misrepresentative of the actual data because it is an absolute representation when uncertainty often exists in the data. While various techniques and tools exist for visualizing uncertainty in scientific visualizations, these do not exist for visualizing information such as graph/network data. With the recent prevalence in data which can be represented as a graph (e.g. social networks), graphs are no longer simple, bi-modal datasets with only nodes and edges. Instead, we are often tasked with working with *multi-modal* graphs where we have multiple types of nodes and edges where each node/edge can have many–perhaps hundreds–of attributes, and these attributes often have some uncertainty attached to them. Moreover, it is often useful to compare multiple graphs of this type as well as the ego networks of nodes in these graphs. In this paper we present various techniques and a prototype tool that can be used to visualize multi-modal graph data with uncertainty attached to each attribute and compare multiple such graphs with one another.

1 INTRODUCTION

One problem with visualization is that it can be misrepresentative of the actual data because it is an absolute representation when uncertainty often exists in the data. While various techniques and tools exist for visualizing uncertainty in scientific visualizations [6] [7] [9], these do not exist for visualizing information, particularly graph network data. With the recent prevalence in data being represented as a graph (e.g. social networks), graphs are no longer simple, bi-modal datasets with only nodes and edges. Instead, we are often tasked with working with *multi-modal* graphs where we have multiple types of nodes and edges where each node/edge can have many–perhaps hundreds–of attributes [12], and these attributes often have some uncertainty attached to them. We can imagine a network having people as nodes whose attributes are age, sex, profession, etc. Edges are associations such as coworker, friend, spouse etc. with attributes such as the number of years the association existed between the two people, number of messages sent/received between the two people, etc. As of now the tool Invenio developed by Singh et al at Georgetown University [12] can be used to visualize multi-modal ($M3$ graphs, see [12]) as seen in figure 3. In this figure there are 3 *node sets* displayed in a traditional node-link diagram, but this is an *absolute* picture. For example, each node could have an attribute called "exists" with uncertainty. This picture would imply that each edge "exists" with total confidence (i.e. zero uncertainty). That is to say, if the edges in the picture did have this attribute with uncertainty, then the picture is misleading. Likewise, with these sorts of uncertain graphs, we could have a graph with the same nodes, but perhaps different edges and different probability distributions for its node and edge attributes, then what would be a good way to compare these two graphs? Also, how do we examining the ego networks (i.e. neighborhoods) of nodes within these graphs as these are often studied in the field of sociology [11]? With these questions in mind, we have the following goals:

- Develop a set of visual techniques which can be used to visualize and compare M^*3 networks with uncertainty attached to node and edge attributes.
- Develop techniques for visualizing and comparing ego networks of nodes in such networks
- Develop a tool which combines all of our techniques to give the user as much information as possible regarding the data and uncertainty in it

In This paper we will discuss the techniques we have developed so far and the prototype of a tool for visualizing and visually comparing these graphs and the ego networks of nodes in these graphs.

Here is an outline of what will be covered:

- Some brief background information
- A review of previous works
- Implementation details
- An in-depth description of our techniques and prototype tool
- Future work and conclusion

2 BACKGROUND

The type of data we want to visualize are multi-modal, multi-relational graphs, or M^*3 graphs/networks as described in [12]. In short, the data can be described as follows:

- $A_x(Id_{A_x}, B_{x1}, \cdots, B_{xr})$
- $E_y(Id_{E_y}, C_{y1}, \cdots, C_{ys})$
- $R_z(Id_{A_x}, Id_{E_y}, D_{z1}, \cdots, D_{zt})$

For example, say we have a network with John, Jane, and Bob. For each person we have the following information: age, sex, profession. In addition we have events, which will be conferences, with information such as the conference name, location, and venue. Each set might look something like the following:

$$\begin{aligned}
A_1(name, age, sex, profession) &= \{(John, 27, Male, sales)\} \\
A_2(name, age, sex, profession) &= \{(Jane, 23, Female, developer), \\
&\quad (Bob, 20, Male, developer)\} \\
E_1(name, size, location, venue) &= \{(IEEEVW, US, Vis)\} \\
E_2(name, size, location, venue) &= \{(S2.0C, US, sales)\} \\
R_1(person, event, attended, published) &= \{(John, S2.0C, 2, 1)\}, \\
&\quad (Jane, IEEEVW, 3, 2)\}
\end{aligned}$$

From the description in Singh et al, relationships can only occur between events and actors. However, in our data, we treat events as just another type of node and allow relationships between actors (nodes). For example, we might have a graph that looks like figure 1.

Two important aspects of the data we wish to visualize are:

1. Two or more graphs which have the same nodes, but possibly different edges between them, but the probability distributions differ between the attributes from one graph to another and
2. The ego networks of nodes within each of these graphs.

In both cases, we wish to visually compare these graphs in a way that will clearly show the differences between the probability distributions.

*e-mail: ncesario@soe.ucsc.edu

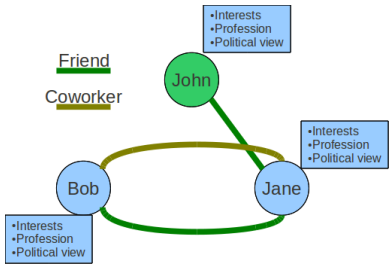

Figure 1: An example of a multi-modal graph with 2 types of nodes and 2 type of edges.

2.1 Synthetic Data

While we have used our techniques to experiment with some real world data, our prototype has primarily been tested on a synthetic dataset generated by Lise Getoor and her team at the University of Maryland. There are 5 node sets, 1 edge set, each node in each node set has approximately 200 attributes and each edge has 4 attributes. For all intents and purposes of this paper we will focus on 5 attributes in the nodes which indicate which set the nodes belong to, and 2 attributes for the edges (one indicates the probability the edge exists while the other is a measure of how similar attributes are between the source and target nodes of a particular edge).

3 Previous Work

While there has been little work done to our knowledge in visualizing uncertainty in graphs and visually comparing uncertainties between multiple M^*3 graphs, there is extensive research in the field of visualizing networks as well as many tools which can be used to visualize a graph/network in a plethora of ways such as:

- Force directed layouts,
- Clustering layouts,
- Radial layouts,
- Various tree layouts,
- etc.

Here we will briefly discuss some of the techniques known to us which are related to our prototype application.

3.1 DualNet

Namata et al [10] developed a method using multiple coordinated views to display network data. In particular, they used multiple views to compare ego networks and used the size and color of nodes to display attributes values.

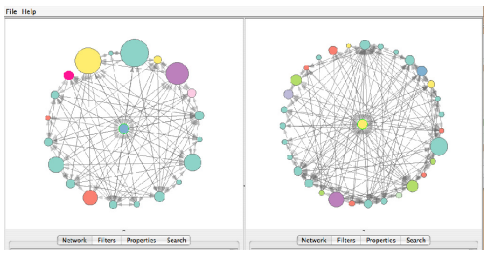

Figure 2: The comparison of two node's ego networks using DualNet [10]

3.2 Invenio

Singh et al [12] developed a tool called Invenio specifically for exploring M^*3 networks. Invenio allows users to interactively color map nodes, run a variety of algorithms such as clustering, look at the network data using several different layout schemes from the Prefuse library, as well as many other features useful for exploring M^*3 networks.

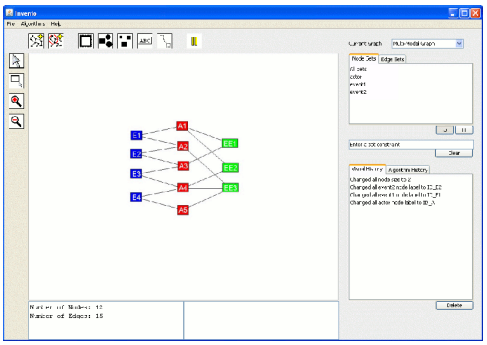

Figure 3: A screenshot of Invenio in action.

3.3 Node Fuzziness

Collins et al [2] use "fuzziness" on nodes to display uncertainty in lattices for decision making. Their application is used to visualize different possible translations for a given phrase, but is could easily be extensible to M^*3 networks. This is a technique we wish to experiment with in our prototype for uncertainty in nodes, but have not implemented yet.

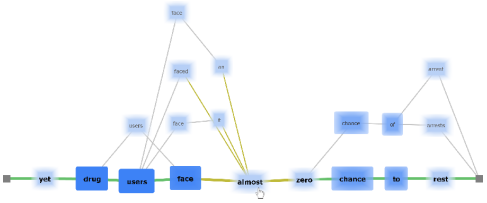

Figure 4: Node fuzziness used to display uncertainty [2].

3.4 Gephi

Gephi [1] is "a tool for people that have to explore and understand graphs. Like photoshop but for data, the user interacts with the representation, manipulate the structures, shapes and colors to reveal hidden properties." (http://gephi.org). While Gephi provides a huge amount of variation for visualizing graph/network data, to our knowledge it does not have any means for exploring M^*3 networks.

3.5 Jigsaw

Gorg et al [3] developed a tool for investigative analysts to explore text data. The multiple displays are used to give the user multiple views of the data which include highlighted textual views, list views, and traditional node-link diagrams.

4 Implementation

We decided to write our tool using the Prefuse [4] toolkit written in Java by Jeffrey Heer at Stanford University which uses Java2D to render.

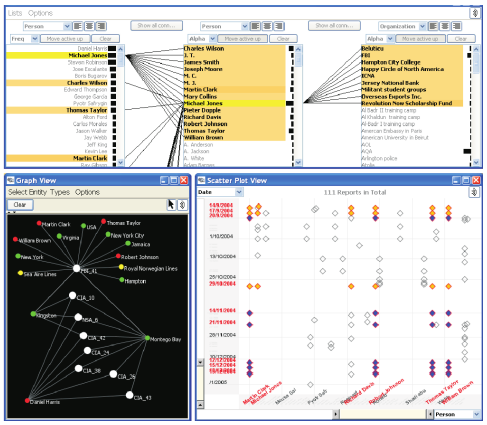

Figure 5: An example of some of the visual techniques used in Jigsaw for visualizing textual data for investigative analysts

Since Prefuse is designed display graph data, much of the functinality we needed for displaying our data is already implemented. Prefuse also uses a table data structure to store nodes and edges. This is ideal for working with M^*3 networks as it allows us to attach an arbitrary number of attribute to nodes and edges. Further, Prefuse is used to visualize data in Invenio, which we are working to integrate our visualizations into as discussed in 6.

5 A Prototype of a Tool

As mentioned earlier, most of the visualizations used in our prototype are variations of techniques previously developed in the visualization community. Our primary contribution is combining several of these modified views to represent M^*3 networks with uncertainty attached to the attributes of nodes and edges in the network.

5.1 Visual Layouts

We will first introduce each visual layout used in the our prototype giving a brief explanation and example of each.

5.1.1 Bullseye

The concept of the bullseye is as follows: given N M^*3 graphs, split the bullseye into N "graph sectors". Further, if each network has n types of nodes, we will split each "graph sector" into n "sub-sectors." At the moment we have only concentrated on instances where $N = 2$ and n 5. Since this is a polar mapping, we can theoretically place an infinite number of graphs on the bullseye with each graph having an infinite number of node types (where infinite is taken to be "practically large"), however, this layout will get cluttered rather quickly as we increase the number of graphs or node types since the graph and sub-sectors will become very small. Still, having a very large number of graphs and/or node types could still be useful with brushing techniques, though we do not explore this here.

Once we find out which sector and sub-sector a node belongs to from its type and graph, we then need to determine its position within the sector. Using polar coordinates, where θ and r are the angular and radial components respectively, we will map θ to a particular attribute's value, and r to the confidence (or 1 - uncertainty) of that attribute's value where $\theta \in (startofsub-sector, endofsub-sector)$.

The radial component, r, will not necessarily be a value from 0 to R where R is the outer radius of the bullseye. Instead, the bullseye can have "rings." Rings can be used for different levels of confidence or, in our case to show which nodes match or don't match between graphs. Within each ring, say r_{ai} is the inner radius of ring a, r_{ao} is the outer radius of ring a, and a node's attribute has uncertainty $u_n = 0.6$. Then the radial component of this node's position is $r_n = u_n(r_{ao} - r_{ai}) + r_{ai}$.

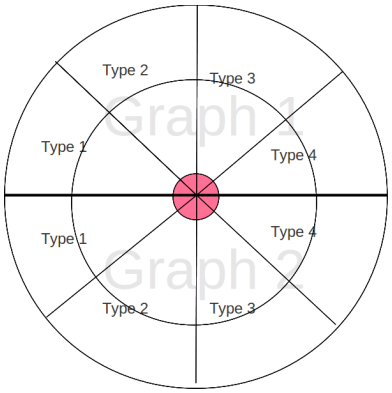

Figure 6: Descriptive diagram of the bullseye layout.

In figure 6, we have a bullseye with 2 graphs, and each graph has 4 types of nodes. In this particular bullseye there are 2 rings. The red circle in the center is used to highlight nodes above a certain probability within the center ring (technically the center "ring" is actually a disc).

With our synthetic data, the main purpose of these rings is to view the comparison of ego networks. We do this with 2 rings. When we compare the ego networks of 2 nodes, the 2 nodes we are comparing and any nodes which are not common between both ego networks appear in the outer ring. Only nodes which are common between both ego networks appear in the center ring/disc. More formally in set notation, if we have a node in graph 1, say A, and a node in graph 2, say B, and the ego networks of A and B are α and β respectively, let $M = \{\alpha \cap \beta\}$, $m = \{(\alpha - M) \cup (\beta - M)\}$. Then all nodes in M appear inside the ring while all nodes in m appear outside of the ring. If no nodes have been selected to compare ego networks (as is the initial state in our prototype), then all nodes appear in the outer ring.

5.1.2 Comparative Column

The bullseye view can be used to compare or display an arbitrary number of graphs. This view, however, is designed to compare exactly 2 graphs. Also different from the bullseye, a node in the Comparative Column (CC) is displayed exactly once, i.e. if the node John appears in both graphs, it is represented as a single node. Similar to a scatter plot, this is a simple rectangular mapping in Euclidean space, except instead of comparing 2 dimensions, we are comparing the uncertainty of a node attribute. Nodes in the center are very similar (e.g. both have uncertainty 0.3), nodes towards the top of the rectangle have a lower uncertainty, and nodes towards the bottom have a higher uncertainty (this could be mapped the average of the 2 nodes' uncertainties or some other measurement of overall confidence between the 2 nodes). If 2 nodes have different uncertainties, then the node will be closer to either the left or right side of the rectangle depending on which graph has lower uncertainty. For example, say some attribute has uncertainty 0.1 in graph one and 0.9 in graph 2, the node will appear much closer to graph one since the attribute has less uncertainty in this model.

5.1.3 Scatter Plot

This is a traditional scatter plot view with nodes as data points and their x-y position based on attribute value vs. uncertainty.

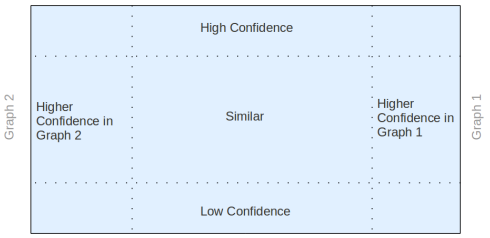

Figure 7: Diagram showing the meaning of each area in the comparative column view.

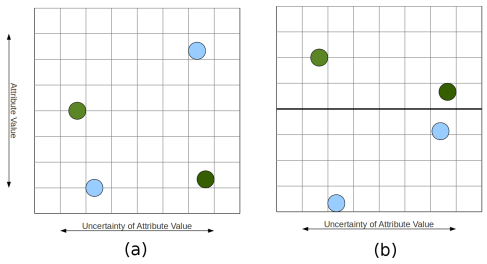

Figure 8: Example of the scatter plot view with (a) nodes not separated by type and (b) nodes separated into vertical "bins."

5.1.4 Fisheye Filter

With any of the techniques aforementioned the display can become cluttered when there are many nodes to visualize. One way to combat this is to use something similar to a fisheye described in [8]. We modified the fisheye algorithm provided by Prefuse slightly to create a sort of "targeted fishey" which only sees nodes of a target type at the focal point while nodes of the non-target type disappear at the focal point, but are visible in the outer edges of the fisheye. We accomplish this by making nodes of the non-target type not visible if they are closer than a certain distance threshold from the focal point of the fisheye. More precisely, if f is the focal point, d is the distance threshold, and A_1 is the target type of node, then

$\forall n \in A_x$
  if $type(n) \neq A_1$
    set n to not visible
  else
    set n to visible

A version of the "targeted fisheye" that uses alpha blending rather than a strict cut-off for visibility is being worked on.

5.1.5 Parallel Coordinates

Parallel coordinates [5] have proven quite useful when it comes to visualizing multidimensional data. In the case of M^*3 graphs, we can treat each actor/event set and edge set as a multi-dimensional space where each attribute is a dimension and the uncertainty of each attribute is its value in that dimension. In figure 10, we see the three nodes mentioned in the background section earlier–John, Jane, and Bob–represented as three poly-lines in a parallel coordinates view. This view tells us that we are very sure of Bob's age, what gender all three people are, and John and Jane's profession while we are not so sure about John's age and Bob's profession.

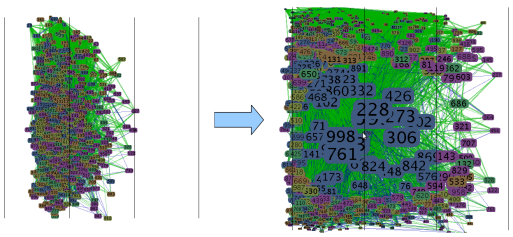

Figure 9: An example of the targeted fisheye. Notice that only blue nodes appear near the center of the fisheye's focal point. It is key to note that the x-y positioning of nodes is only preserved at the focal point of the fisheye. In all images of the prototype, edges thickness is mapped to the probability an edge exists (thicker means higher probability of existence) while color is mapped to a numerical score representing how similar node attributes are between the source and target nodes of a given edge (blue is a very low score, green is a high score).

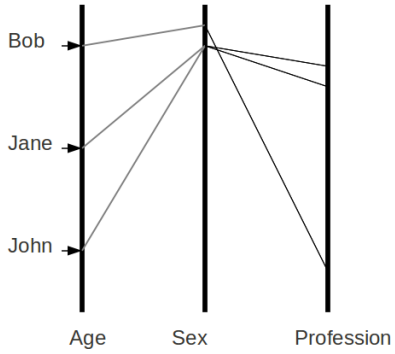

Figure 10: Parallel coordinates used to visualize the nodes in a multimodal graph.

5.2 Linked Views

Each of the visual techniques mentioned in this section give drastically different views from one another and, with the exception of parallel coordinates, are designed to only display/compare 1 or 2 node attributes. Further, the nodes and edges in our data will often contain 10s (if not much more) of attributes. For this reasons, our tool needs the ability to have many displays with the option to link them together in order to make these many different "vantage points" of the data useful to the user.

6 Future Work

As this is a prototype and a first approach at this problem, there are many features we would like to add and experiment with.

6.1 Visualizing Edge Types and Attributes

As of now only the thickness, color, and opacity of edges have been used to map the type and attributes of an edge. However, as is apparent in figure 9, these mappings are not useful when there are many edges. Since the majority of techniques developed thus far are focused on nodes, we are currently experimenting with representing edges as nodes and using some of the techniques we have presented in this paper.

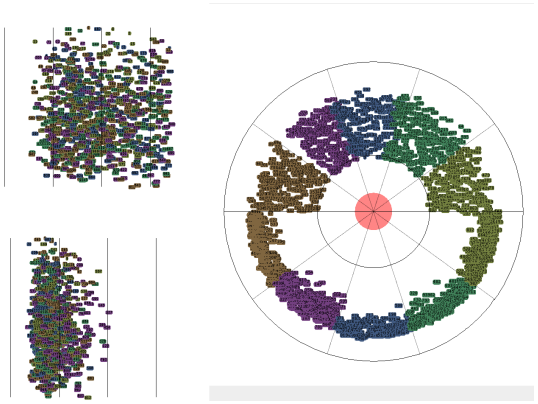

Figure 11: An example of linked displays in our prototype tool. Here we want to compare 2 multi-modal graphs. We are using 2 "organized" scatter plot views to compare 2 attributes in each graph separately and then one bullseye view to show yet another comparison of attribute uncertainties. Edges are hidden to help concentrate on the nodes in this particular example.

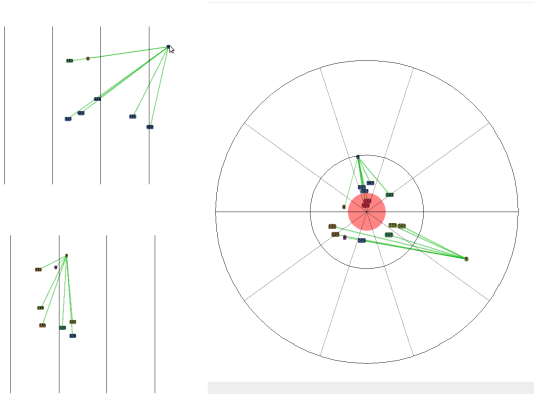

Figure 12: Comparing ego networks from the same graphs as in 11. By clicking on a node (in this case node "9") in, say, graph 1, we compare node "9"'s ego network in graph 1 with node "9"'s ego network in graph 2.

6.2 Allow User to Add/Remove and Link/Unlink Displays

As of now the number of displays is fixed and all displays are linked. Ideally the user should be able to add as many displays as wanted, or as many as is suitable for his or her screenspace. The user should be able to experiment with all techniques, simultaneously if desired, and remove displays which do not appear to be beneficial.

On a similar note, linked displays can be very useful, however, there may be cases where the user wishes to explore different parts of a dataset simultaneously with multiple displays. This is currently not possible as all displays are linked by default, therefore whatever data is being examined in one display is also the focus of all the other displays.

6.3 Details-on-Demand

As can be seen in many of the example figures, the displays can become cluttered quite easily and it is often desirable to display a single node as a simple shape with no information, except for color,

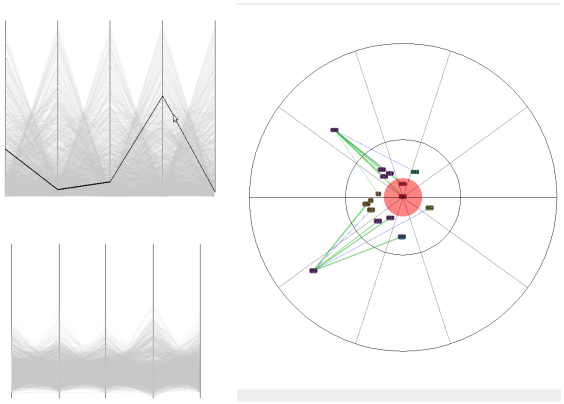

Figure 13: Here we are again using the same graphs as in figures 11 and 12, but now instead of the scatter plot views we are using the parallel coordinate views to look at each graph individually. In this particular instance we have selected a node from the top graph by selecting this node's corresponding poly-line in in the parallel coordinates view.

perhaps. In the example images displayed here, notice that there is only a number (the node's ID) displayed on each node, there are actually hundreds of attributes associated with each node. For this reason, the user should be able to immediately see all or a large portion of this data upon selecting a node.

6.4 Animation

While the prototype is interactive, nothing is animated. It would be helpful to give the user contextual information when he or she changes the display. For example, if the user changes the layout used in a display, if nodes "migrate" to their new positions rather than just appear there, it would give the user an idea of how one layout relates to another and possibly help them to understand the data better.

6.5 Integration with Invenio

We are currently working with Lisa Singh and her team at Georgetown University to integrate some of the layouts discussed here into their tool Invenio. Invenio already has functionality for the user to add and remove displays as well as run many different graph algorithms and read in a variety of graph formats. Further, its focus is on M^*3 graphs. See [12] for more details.

6.6 User Evaluation

As mentioned earlier we have some real world data, but we are still developing visual techniques using synthetic data. After getting our prototype to a user-worthy state, we plan to complete a user evaluation. Our primary focus will be on social network data and looking at what our tool exposes. In addition, it would be helpful to look at data which contains uncertainty and then view it using traditional tools which do not take uncertainty into account and contrast those results with results from using our tool.

7 Conclusion

In this first attempt at visualizing uncertainty in graphs, comparing these uncertain graphs, and visualizing/comparing the ego networks in such graphs, we have presented a prototype which compiles 5 different visual techniques which can be linked together to better understand the data. These techniques were: bullseye (5.1.1), comparative column (5.1.2), scatter plot (5.1.3), fisheye (5.1.4), and parallel coordinates (5.1.5). While the tool we have developed to use

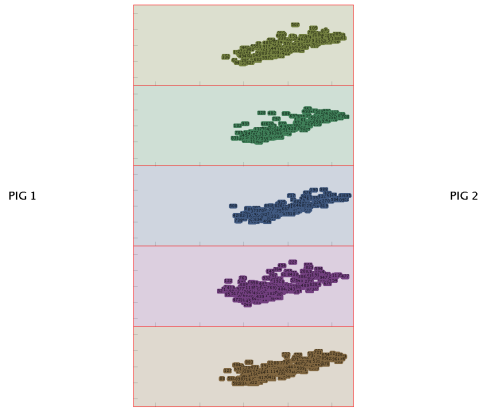

Figure 14: Example of the comparative column view using the same 2 graphs as in the previous figures. The fact that the majority of nodes are "leaning" to the right means most of the nodes have a less uncertainty in PIG2 (PIG stands for Probabilistic Information Graph). In this particular view, we are actually using 5 comparative columns (one for each node type). See section 5.1.2 for more details.

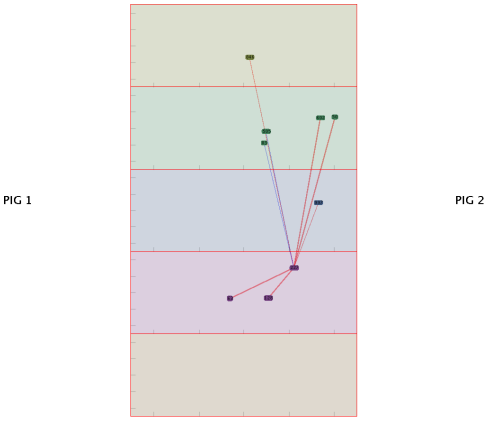

Figure 15: Using the comparative column view to look at the ego network of a node.

all of these techniques is a somewhat limited prototype, it can still be used to view M^*3 networks in ways which could not be done before with an emphasis on uncertainty in node and edge attributes.

REFERENCES

[1] M. Bastian, S. Heymann, and M. Jacomy. Gephi : An open source software for exploring and manipulating networks, 2009.

[2] C. Collins, S. Carpendale, and G. Penn. Visualization of uncertainty in lattices to support decision-making, 2007.

[3] C. Gorg, Z. Liu, N. Parekh, K. Singhal, and J. Stasko. Visual analytics with jigsaw. In *VAST '07: Proceedings of the 2007 IEEE Symposium on Visual Analytics Science and Technology*, pages 201–202, Washington, DC, USA, 2007. IEEE Computer Society.

[4] J. Heer. http://prefuse.org.

[5] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *VIS '90: Proceedings of the 1st conference on Visualization '90*, pages 361–378, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.

[6] D. Kao, J. L. Dungan, and A. Pang. Visualizing 2d probability distributions from eos satellite image-derived data sets: a case study. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 457–460, Washington, DC, USA, 2001. IEEE Computer Society.

[7] D. Kao, A. Love, J. L. Dungan, and A. Pang. Picturing data with uncertainty. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Posters*, page 104, New York, NY, USA, 2004. ACM.

[8] Y. K. Leung and M. D. Apperley. A review and taxonomy of distortion-oriented presentation techniques, 1994.

[9] S. K. Lodha, A. Pang, R. E. Sheehan, and C. M. Wittenbrink. Uflow: visualizing uncertainty in fluid flow. In *VIS '96: Proceedings of the 7th conference on Visualization '96*, pages 249–ff., Los Alamitos, CA, USA, 1996. IEEE Computer Society Press.

[10] G. M. Namata, B. Staats, L. Getoor, and B. Shneiderman. A dual-view approach to interactive network visualization. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 939–942, New York, NY, USA, 2007. ACM.

[11] J. Scott. *Social Network Analysis*. 1987.

[12] L. Singh, M. Beard, L. Getoor, and M. B. Blake. Visual mining of multi-modal social networks at different abstraction levels. In *In IEEE Conference on Information Visualization*, 2007.