Revisited Experimental Comparison of Node-Link and Matrix Representations

Mershack Okoe[1], Radu Jianu[2], and Stephen Kobourov[3]

[1] Department of Computer Science, Florida International University, Miami, USA
mokoe001@cis.fiu.edu
[2] giCentre, City, University of London, UK
radu.jianu@city.ac.uk
[3] Department of Computer Science, University of Arizona, Tucson, USA
kobourov@cs.arizona.edu

Abstract. Visualizing network data is applicable in domains such as biology, engineering, and social sciences. We report the results of a study comparing the effectiveness of the two primary techniques for showing network data: node-link diagrams and adjacency matrices. Specifically, an evaluation with a large number of online participants revealed statistically significant differences between the two visualizations. Our work adds to existing research in several ways. First, we explore a broad spectrum of network tasks, many of which had not been previously evaluated. Second, our study uses a large dataset, typical of many real-life networks not explored by previous studies. Third, we leverage crowdsourcing to evaluate many tasks with many participants.

1 Introduction

Visualizing network data is known to benefit a wide range of domains, including biology, engineering, and social sciences [54]. The data visualization community has proposed many approaches to visual network exploration. By comparison, the body of work that evaluates the ability of such methods to support data-reading tasks is limited. We describe the results of a comparative evaluation of the two most popular ways of visualizing networks: node-link diagrams (NL) and adjacency matrices (AM). Specifically, we consider two interactive visualizations (NL and AM), using a crowdsourced, between-subject methodology, with 557 distinct online users, 14 evaluated tasks, and 1 real-world dataset; see Fig. 1.

Several earlier studies compare NL and AM visualizations on specific classes of networks and using a variety of tasks [21, 22, 36, 30]. They show that the effectiveness of the visualization depends heavily on the properties of the given dataset and the given data-reading tasks. For example, Ghoniem *et al.*'s seminal evaluation [21] found that the two visualizations' ability to support specific tasks depends on the size and density of the network. Similarly, it is reasonable to hypothesize that there might be differences depending on the structure of the network (e.g., clustered networks, small-world networks). Thus exploring the
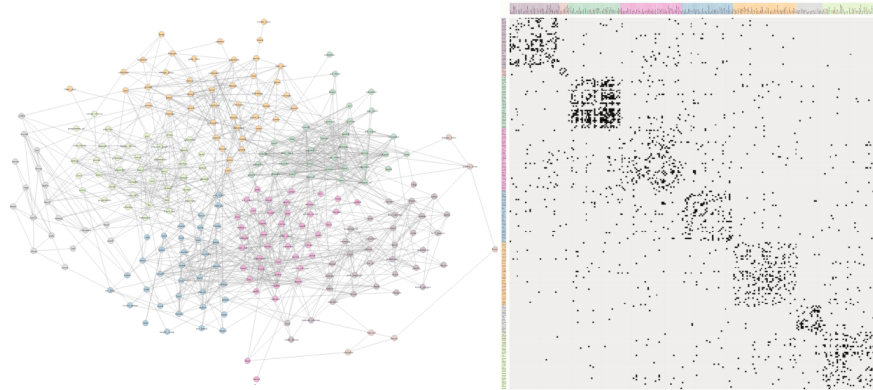
Fig. 1. Evaluated visualizations: node-link diagram and adjacency matrix.

effectiveness of NL and AM visualizations on different types of graphs, and using a broader spectrum of tasks, seems worthwhile.

Our study uses one real-world, scale-free dataset of 258 nodes and 1090 edges. This makes our dataset different in structure and larger than previously evaluated networks. For example, Ghoniem *et al.* evaluated random networks that were about 2.5 times smaller, albeit somewhat denser. We argue (in section 3) that our chosen dataset is worth studying as it exemplifies a large class of networks that occur in real applications.

More recently, networks are used to solve increasingly complex problems and as a result, there is an expanding range of tasks that are relevant in real applications and which are of interest to the visualization community. Our study evaluates many tasks (14), carefully chosen to span multiple task taxonomies [32, 4]. Many of these tasks were not previously investigated in the context of NL and AM representations.

Given the caveat that these results apply to the specific underlying network and the specific implementations of NL and AM visualizations, some of our results confirm prior observations in similar settings, while others are new. NL outperforms AM for questions about graph topology (e.g., "Select all neighbors of node," "Is a highlighted node connected to a named node?"). Of 10 such tasks, participants who used the node-link diagram were more accurate in 5 and less accurate in 2. NL and AM give similar results for 4 tasks which tested the ability of the participants to identify and compare node groups or clusters, except one instance in which AM outperforms NL. Finally, NL and AM provide similar results on 2 memorability tasks. The full results are shown in Figure 4.

2   Related Work

Considerable effort has been expended on optimizing NL and AM visualizations to remove clutter, increase the saliency of visual patterns, and support data

reading tasks [54]. NL, AM, and slight variations thereof have long been used in practice to support analyses of data in a broad range of domains, including proteomic data [50, 29, 8, 28], brain connectivity data [3], social-networks [53], and engineering [49].

Static visual encodings were augmented by interaction to support the exploration and analysis of large and intricate datasets typical of real-life applications. Interactive systems that visualize complex relational data use NL [6, 9, 50], and AM [20, 10, 7, 12, 45, 11, 17, 51]. We reviewed such systems to determine common interactions and included them in our evaluated visualizations.

While the two types of visualizations have been used broadly for a long time, studying how people parse them visually and which visualization method better supports specific tasks and datasets, is ongoing. For example, studies by Purchase *et al.* [40, 55, 41] consider how node-link layouts impact data readability, eye-tracking research by Huang *et al.* reveal visual patterns and measure the cognitive load associated with network exploration [26, 27]. More recently Jianu *et al.* and Saket *et al.* consider the performance of node-link diagrams with overlaid group information [28, 48].

Our work is one in a series of studies that compare NL and AM representations. Ghoniem *et al.* [21] evaluated the two approaches on seven connectivity and counting tasks, using interactive visualizations (e.g., node can be selected and highlighted). Synthetic graphs of three sizes (20, 50, 100 nodes) and three densities (0.2, 0.4, and 0.6) were used. The authors found that for small sparse graphs, NL was better in connectivity tasks, but that for large and dense graphs, AM outperformed NL for all tasks. Similarly, Keller *et al.* [30] evaluated six tasks on three real-life networks of varying small sizes (8, 22, 50) and three densities (unspecified, 0.2, 0.5). Using both static and interactive variants of NL and AM, Abuthawabeh *et al.* found that the participants were equally able to detect structure in graphs representing code dependencies [1]. Alper *et al.* found that in tasks involving the comparison of weighted graphs, matrices outperform node-link diagrams [3]. Finally, Christensen *et al.* [16] evaluated matrix quilts in addition to NL and AM in a smaller scale study.

Our study adds to what is already known in several ways. First, we explore a significantly broader range of tasks than earlier studies. These were carefully selected to cover the graph task taxonomy of Lee *et al.* [32] and the general taxonomy of visualization tasks by Amar *et al.* [4]. We also considered the task taxonomies for simple graphs [32], clustered graphs [47], and more generally for visualization tasks [4, 52], which have been found to be useful in guiding research and informing user study task choices [28, 48]. Second, our study uses a large real-world network, typical of many scale-free networks that arise in practical applications. Finally, unlike previous studies, we leverage crowdsourcing, via Amazon's Mechanical Turk, to evaluate many tasks with many participants.

Note that Mechanical Turk provides access to a diverse participant population [33, 31], and is considered a valid platform for evaluation in general [38, 31], as well as specifically in the context of visualization studies [23]. Many recent visualization studies are crowdsourced [15, 35, 28, 43, 14] and specific platforms

for online evaluations are developed, including GraphUnit designed for online evaluation of network visualizations [37].

3   Study Design

3.1   Stimuli: Data

We evaluated a single network with 258 nodes and 1090 edges, representing cooking ingredients connected by edges when frequently used together in recipes. The density of the network was 0.016 (computed as $\#edges/\#nodes^2$). This network had been explored previously by Ahn *et al.* [2]. In its original form, the network is larger (381 nodes) but we reduced it slightly to ensure it could be visualized smoothly in a browser. We did so by removing disconnected components and low-weight edges. Evaluating a single dataset allowed us to cover a broad spectrum of tasks while keeping the size of the study manageable, but naturally, this choice has several limitations, discussed in section 5.

Rationale: Our motivation for choosing our network was three-fold. First, it is *different than those evaluated already.* Our network is 2.5 and 5 times larger than those evaluated by Ghoniem *et al.* and Keller *et al.*. Second, our network was chosen as a *representative of several types of real-world networks.* Specifically, we reviewed 17 relational datasets (e.g., trade exchanges between countries, the Les Miserable dataset, TVCG paper co-authorships, protein-interaction networks). We selected one from this set that was representative in terms of structure and density, while at the same time sufficiently small to be evaluated in a browser. Our network has about 4 times more edges than nodes. This was close to the average edge/node ratio in the 17 networks we reviewed and representative of many networks commonly found in practice [34]. Third, we believe a dataset revolving around cooking ingredients would have a *greater appeal to participants.* Ingredients were shown as node labels and several tasks referred to ingredients by name. Relatable, concrete dataset may help users understand tasks better [5].

3.2   Stimuli: Visual Encoding

We evaluated two visual encodings: a node-link diagram (NL) drawn using the neato algorithm from graphviz [18], and an adjacency matrix (AM), sorted to reveal clusters using the barycenter algorithm available in the Reorder.js library [20]. We clustered the network using modularity clustering from GMap [25] and encoded this information in the two visual representations using color, as shown in Fig. 1. Both visualizations were developed using the D3 web-library.

Rationale: The neato algorithm is provided in popular layout tools such as graphviz and frequently part of NL evaluations [21, 28]. We ordered our AM to reveal structure, as we considered this more representative of how matrices are used in practice, unlike Ghoniem *et al.* [21], who used a lexicographical order.

3.3   Stimuli: Interactions

Both visualizations support panning and zooming, using the mouse-wheel. Multiple nodes can be selected by clicking on them, and deselected with an additional click. Selecting a node in NL colors both the node and its outgoing edges in purple. Selections in AM operate on node labels but change the color of the corresponding node's row or column. Similarly, node mouse-over in NL turns the node and its edges green and shows the node label via tooltips. Node mouse-over in AM colors the row or column. Note that for both node selection and node mouse-over in AM, if a row (column) is colored the complementary column (row) is not. We chose this approach since both Ghoniem *at al.* and Okoe *et al.* mention that multiple markings for the same node can confuse users [21, 28].

To select a node as the answer to a task, the participants double-click it. This marks the node with a thick black contour. In both NL and AM this marking was restricted to nodes and labels, without extending to edges or rows/columns. The participants could also deselect an answer by double-clicking it again.

Similar interactions apply to edge selection: An edge mouse-over in NL turns the edge green, and if clicked it is selected and so turns purple. In AM, hovering over an edge-cell highlights its corresponding row and column in green, and clicking it selects the edge.

Rationale: We chose to evaluate interactive visualizations as interactivity is typical in real-world applications. Previous studies, such as those of Ghoniem *et al.* or Keller *et al.*, also used basic interactions for the same reason. Interactivity can significantly change the effectiveness of a visual encoding, however, and a careful choice of interactive techniques is warranted.

Our goal was to use interactions that are *ecologically valid* (i.e., representative of interactions typical of NL or AM visualizations) and *fair* (i.e., providing similar functionality and power in both visualizations). To this end, we reviewed 9 systems for network visualization (e.g., Gephi [9], Cytoscape [50], Tulip [6]), 12 network evaluation papers (e.g., Ghoniem *et al.*[21], Keller *et al.* [30], Okoe *et al.*[36]) and 6 systems and papers for adjacency matrices (e.g., ZAME [19],TimeMatrix [56], work by Perin *et al.* [39], work by Henry *et al.* [24]). We cataloged the interactions described or available in these systems, as well as their particular implementation, and then selected the set of most common interactions.

This resulted in a set of interactions that both overlapped and differed slightly from those implemented in previous studies. Overlapping interactions were described above. New interactions included zooming and panning, which was required to solve some of the tasks. We believe the addition of zooming and panning is valuable since such basic navigation is an integral part of real-life systems. Our node-link diagrams also allowed users to move nodes, an interaction that can be used to disambiguate cases in which nodes or edges overlap, and is ubiquitous in NL systems. This interaction does not have an equivalent in AM but is also not necessary as rows and columns are evenly spaced.
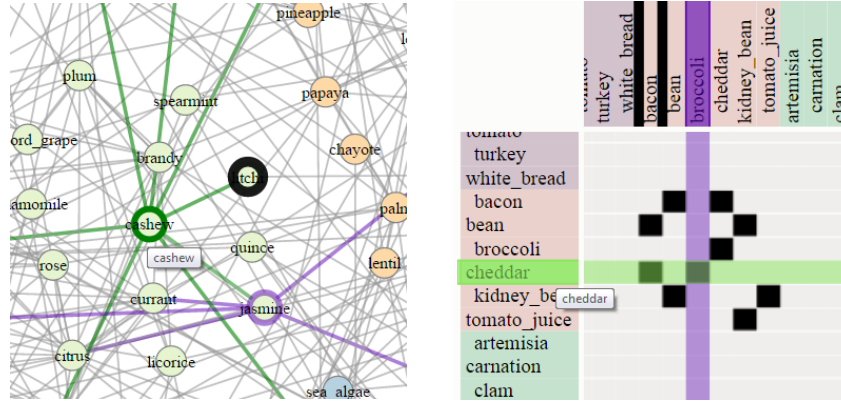
Fig. 2. Participants mouse-over nodes to highlight them (green) and click on nodes to select them (purple). Designating a node as the answer for a task answer is accomplished via a double-click, which draws a black contour around the node.

3.4   Tasks

We evaluated the 14 tasks described in Table 1. Participants solved multiple repeats (generally 5 or 10) of each task. Task repeats were selected manually on the network so as to cover multiple levels of complexity. For example, our repeats included nodes with both low and large degrees (e.g., $T1$, $T2$), short and long paths (e.g., $T10$, $T13$), or nodes with few and many neighbors (e.g., $T4$).

Three of our tasks warrant a more detailed discussion. We included two memorability tasks, ($T11$, $T14$). The former tested the ability of participants to recall data they had looked for or accessed at an earlier time, and is similar to memorability tasks evaluated by Saket *et al.* [46]. The latter tested the ability of participants to recognize visual configurations they had viewed previously and is more similar to tasks used by Jianu *et al.* and Borkin *et al.* [28, 14]. Both memorability tasks were based on questions that the participants had to answer early in their session (i.e., $T9$ in group 4, and $T12$ in group 5) to prime the participants with a particular piece of information or visual configuration. A few minutes later, after performing a set of other tasks (i.e, $T10$ in group 4, $T13$ in group 5), the participants were asked about the information from the earlier task. Finally, we added a path-estimation task ($T5$), which required the participants to estimate how far two nodes are, in terms of the shortest path between them. Timing constraints ensured that participants used perceptual mechanisms to give a best-guess response instead of "computing" the correct answer.

Rationale: Our overarching goal in selecting our tasks was to cover a wide spectrum of different and realistic network tasks. We selected tasks to cover the graph objects they provide answers about (i.e., nodes, edges, paths), as well as cover Lee *et al.*'s categories of graph-reading tasks, and Amar *et al.*'s general types of visualization tasks. Several of our tasks have been used before but under slightly different conditions. Additionally, we included tasks that go

Task	Target	Task tax. [32]	Task tax. [4]	Group	#Repeats	Time
1. Given two highlighted nodes, select the one with the larger degree.	node	Topology (adjacency)	Retrieve value, Sort	1	10	15
2. Given a highlighted node, select all its neighbors	edge	Topology (adjacency, accessibility)	Retrieve value, Filter	1	10	25
3. Given two clusters of highlighted nodes, which one is more interconnected?	clusters, cliques	Overview (connectivity)	Filter, Sort, Cluster	1	10	10
4. Given two highlighted nodes, select all of the common neighbors.	edge	Topology (shared neighbor)	Retrieve value, Filter	2	10	30
5. Given two pairs of highlighted nodes (red and blue) and limited time, estimate which pair is closer in terms of graph topology?	path, edge	Overview (connectivity)	Derive value, Sort	2	10	10
6. How many clusters are there in the visualization? *clusters shown via color (section 3.2)	clusters	Overview (connectivity)	Derive value	3	1	10
7. Given two groups of highlighted nodes (e.g., red and blue) and limited time, estimate which group is larger.	clusters	Attribute based	Filter, Sort, Derive value, Correlate	3	10	10
8. Given two highlighted nodes decide whether they belong to the same cluster. *clusters shown via color (section 3.2)	clusters, nodes	Attribute based	Cluster, Filter	3	10	10
9. Given one highlighted node and one named node, are they connected?	edge	Topology (adjacency)	Retrieve value	4	5	20
10. Given two highlighted nodes, how long is the shortest path between them?	path, edge	Topology (connectivity)	Retrieve value, Derived value, filter	4	5	60
11. Memorability: After spending several minutes on task 10, can participants remember the answers they gave to task 9, without access to the visualization?		See section 3.4	See section 3.4	4	5	unlim
12. Given two highlighted nodes and three named ones, which of the named nodes is connected to both highlighted nodes? (exemplified in Figure 3)	edge	Topology (shared neighbor)	Retrieve value, Filter	5	5	60
13. Given a selected node, how many nodes are within two edges' reach?	edge	Topology (accessibility)	Retrieve value, Derive value, Filter	5	5	60
14. Memorability: After spending several minutes on tasks 13, can participants remember (i.e., select) which nodes were highlighted as part of task 12, if showed the visualization with the answers they gave to task 13 highlighted?		**See paper body	**See paper body	5	5	unlim

Table 1. Tasks: the columns describe (i) the task, (ii) targeted network element, (iii-iv) task categories in Lee *et al.*'s and Amar *et al.*'s taxonomies, (v) group number the task was evaluated in, (vi) number of instances of this task type, (vii) task time limit (sec).

beyond previous studies comparing NL and AM, such as tasks involving clusters. We also included memorability tasks as they are a topic of growing interest in the visualization community [14, 46]. We also hypothesized there would be differences between the two visualizations in this respect. We included a path-estimation task [28], as it is a good representative of the "Overview" category of graph tasks, and underlies perceptual queries that users make on relational data.

3.5 Hypotheses

Based on previous results by Ghoniem *et al.* [21], Keller *et al.* [30], Okoe *et al.* [37], Jianu *et al.* [28], and Saket *et al.* [48] we devised the null hypotheses:

H1: There is no statistically significant difference in time and accuracy performance between using NL and AM for tasks involving the retrieval of information about nodes and direct connectivity ($T1$, $T2$, $T4$, $T9$, $T12$).

H2: There is no statistically significant difference in time and accuracy performance between using NL and AM for connectivity and accessibility tasks involving paths of length greater than two ($T5$, $T10$, $T13$).

H3: There is no statistically significant difference in time and accuracy performance between using NL and AM on group tasks ($T3, T6, T7, T8$).

H4: There is no statistically significant difference in memorability between using NL and AM.

We expected H1 to hold and H2 not to hold. We also thought H3 would hold, except for estimating group interconnectivity ($T6$), since estimating the number of non-overlapping dots in a square (AM) should be easier than estimating overlapping edges in an irregular 2D area (NL). Finally, we anticipated memorability would be higher in node-link diagrams due to its more distinguishable features.

3.6   Design

We used a between-subjects experiment with two conditions. We divided our 14 task types into 5 experimental groups, as shown in Table 1, and we evaluated each group separately. Each participant was allowed to participate in a single group and used just one of the two visualizations. We assigned participants to groups and conditions in a round-robin fashion. We aimed to collect data from around 50 participants per condition. As some participants did not complete the study, the total number of participants for whom we collected data varies slightly between conditions. All tasks were timed as shown in Table 1, with time limits determined experimentally through a pilot-study and chosen to allow most participants to complete the tasks, while moving the study along.

Rationale: Between-subject experiments are frequently used in the visualization community [28, 48, 57, 13, 42, 31, 35]. One advantage of this design is the absence of learning effects between evaluated conditions. A disadvantage is the need for large numbers of participants, which is easily mitigated in a crowdsourced setting. Moreover, between-subjects designs are quicker (since only one condition is evaluated at a time) and online participants prefer shorter studies.

We divided the tasks into groups for the same reason. Having each participant evaluate all tasks would have resulted in excessively long sessions that participants would have found tiring. Having participants solve only subsets of tasks allowed us to reduce their time commitment. We used estimated task completion times to group tasks, aiming for an expected duration of about 15 minutes.

We aimed for 50 participants per condition, matching the numbers used in earlier crowdsourced studies [15, 28]. We decided to enforce short time-limits in order to limit and make uniform the total session duration across participants.

3.7   Procedure

We used Amazon's Mechanical Turk (MTurk) to crowdsource our study to a broad population. To account for variations in participant demographics during the day, we published study batches throughout the day. We ran conditions in parallel and directed incoming participants to them using a round-robin assignment, to ensure that the two conditions sampled participants from the same populations. The demographics of MTurk users are reported by Ross *et al.* [44].

Each incoming participant was first presented with an introduction to the study, dataset, the visualization they would see and use, and the tasks they would perform. Each task was exemplified in the introduction, as shown in Figure 2. Since our interactions relied on color, participants were administered a color-blindness test. Next came a training session which involved solving two instances of each type of task in their assigned group. During the training session the participants could check the correctness of their answers.

Finally, the participants were lead to the main part of the study. In the main part of the study, task instances of each type in an assigned group were shown to the participants. For example, since group 1 involved three distinct task types, participants assigned to it solved three consecutive sections of ten task-instances each. At the end, we asked the participants for comments.

We used GraphUnit [37] to create the study, deploy it, and collect data. Visualizations were shown on the left, while task instructions and answer widgets were shown on the right. Depending on each task, users answered by selecting nodes or by using interactive widgets (e.g., text-boxes, check-boxes). Time limits were enforced by showing a count-down timer and hiding the visualization once the counter expired. To increase the chances of collecting clean data we awarded a bonus to the best result in each group and told participants that two of the task-instances were control tasks easy enough for anyone to solve.

4   Results

Our results are summarized in Fig. 4. By and large, they show that node-link diagrams were better for most types of connectivity tasks ($T1$, $T2$, $T4$, $T5$, $T9$, $T10$, $T13$) thereby invalidating both H1 and H2. The fact that H1 does not hold is surprising given previous results. Performance on group tasks was generally comparable with the two visualizations, as hypothesized ($H3$), though we found that the AM was better for estimating the number of clusters rather than their interconnectivity. Finally, NL supported memorability tasks better (invalidating $H4$). In particular, NL users outperformed AM users when recalling previously used data ($T11$).

Data processing: We collected data from 557 individual participants distributed across task groups and conditions as shown in Fig. 3. We removed a total of 28 responses from participants who spent an average of 2 seconds per

Groups	Condition	User Size	Valid Data
1	NL	65	63
	AM	62	58
2	NL	58	54
	AM	53	50
3	NL	55	53
	AM	55	52
4	NL	52	50
	AM	53	50
5	NL	54	52
	AM	50	47

Fig. 3. Number of participants in each task group per condition and the number of valid submissions used after data cleaning.

task and had accuracy in the bottom 10 percentile. We considered these likely to be random responses by participants attempting to game the study.

To compute the accuracy of node selections ($T1$, $T2$, $T4$), we used the formula $Acc = (\|PS \cap TA\|)/\|TA\|\}$, where PS is the participant's selection and TA is the true answer. To compute answers for tasks involving numeric answers ($T6$, $T10$, $T13$) we used the formula $Acc = max(0, 1 - \|PA - TA|/|TA|)$, where PA is the participant's answer and TA is the true answer. For other tasks we gave a 1 to correct answers, and a 0 to incorrect answers. Since each task type was represented in the study by several repeats, we averaged the accuracies of a task's individual repeats into an accuracy for the task as a whole.

Statistical analysis: If the data is normally distributed (determined via a Shapiro-Wilk test) we use a t-test analysis between conditions to determine if the observed differences are significant. Otherwise we use a Wilcoxon-Rank-Sum test. We indicate statistically significant differences and effect sizes in Fig. 4.

5   Discussion

Based on the quantitative results and our own interactions with the visualizations, we believe the results can be explained by several factors.

First, NL can be more compact than AM since their layout fully leverages the 2D area, while matrices are constrained to two 1D linear node orders. Matrices favor dense networks (as number of edges increases, matrix size remains constant) but not sparse ones (empty matrices are as large as a dense ones). Instead, sparse NL diagrams can be packed tightly. At the extreme, an empty network can be shown without loss in readability using NL in a $\sqrt{N} \times \sqrt{(N)}$ square. The same empty network would require a $N \times N$ square in an AM. Thus, as networks grow larger but not necessarily denser, AM may incur an increasing navigation cost. Concretely, our NL diagrams required less zooming for nodes to become legible and selected accurately. This could explain the differences in $T1$.
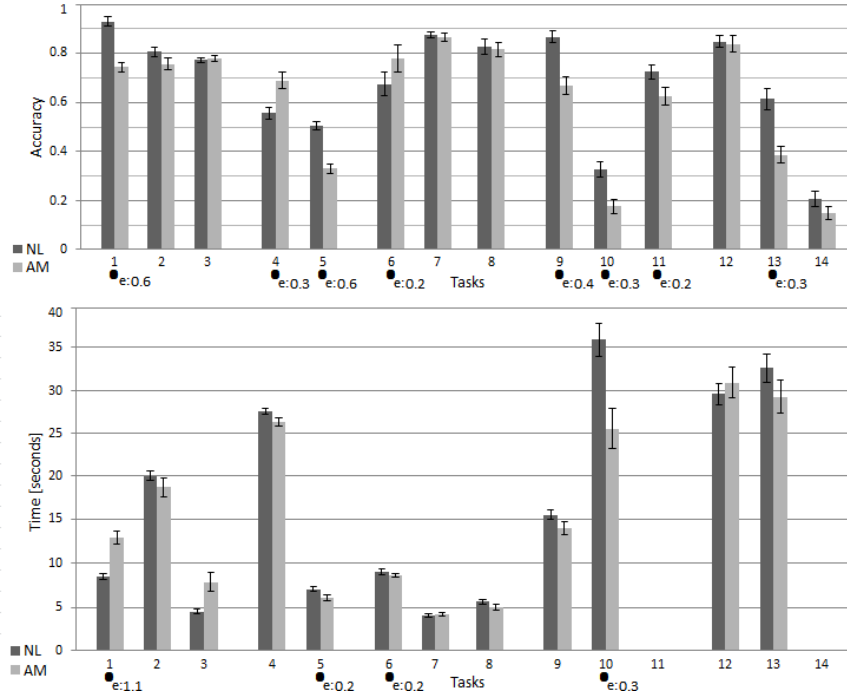
Fig. 4. Results: accuracy and time. Error bars show one standard error. Statistically significant results and effect sizes are also marked. Tasks 14, 11 had no time limits.

Second, NL draw a node's glyph and connections together. Thus, once a label is spotted, from it, its outgoing edges can be traced to other nodes and their labels. Moreover, the presence of the edge aids this tracing. Instead, matrices show node information and edge information separately. Finding the endpoints of an edge involves two potentially long visual-traces along the horizontal and vertical axes. Similarly, finding an edge of an identified node involves a horizontal or vertical search. This could be one of the reasons for the large effect in $T9$. However, this described behavior is only hypothesized and yet to be demonstrated.

Third, Ghoniem *et al.* found that AM performs poorly on tasks involving long paths [21], and our results on $T10$ and $T13$ confirm this. Interestingly, the average time of participants performing path tasks ($T10$) in AM is significantly shorter than that for NL. However, we found that this is due to many AM users giving up on solving the task altogether early on. Moreover, NL layouts aim to place nodes so that their network distance matches their embedded distance. While matrices can also order rows and columns, they are constrained by the use of a single dimension. This could explain the results of $T5$: when one pair of nodes were in the same cluster and the other not, comparing their topological proximity was possible in both visualizations, but in all other cases NL outperforms AM.

Matrices eliminate occlusion and ambiguity problems. In NL diagrams it is sometimes difficult to tell if an edge connects to a node or passes through it, but this is not the case in AMs. Moreover, many tasks that involve visual searches in unconstrained 2D space with NL, are easier with AM. For example, finding a node in an AM involves a linear scan in a list of labels. Counting nodes with certain properties can also be done sequentially by moving through the matrix's headers. Such tasks are difficult in NL diagrams as users have to search a 2D space and keep track of already visited nodes. This may account for $T4$, where AM outperforms NL: participants could systematically scan two selected AM node-rows and identify the columns where both rows had an edge.

Limitations: Several earlier studies comparing NL and AM considered the effects of network size and density [22, 30]. While we recognize the value of this approach, this was beyond the scope of our current study. Instead, we aimed to understand how the two visualizations support a more complete range of tasks (14 versus previously 7 and 6) in a network that is representative of real-world networks in size and structure. It is unclear whether our results would generalize to real-world networks that are significantly larger or denser but our work does provide additional experimental data for a network unlike those evaluated earlier.

We use one type of network and a single instance thereof. This is a methodological drawback which we accepted, due to the overhead associated with preparing multiple appropriate real-world networks for evaluation and phrasing participant instructions using the semantics of different networks. While the limitations of this approach are non-trivial, we attempted to balance them by using multiple task-repeats of the same type and focusing on different parts of the network.

The density of our network was significantly lower than [21, 30]. However, Melancon points out that large real-world networks with high densities are rare [34]. He argues that the edge-to-node ratio is a better indicator for density in real-world networks as it is less sensitive to the number of nodes. Indeed, only 1 of the 17 networks we considered, and 3 of the 19 networks Melancon considered had densities higher than 0.2. In 3 of these 4 cases, these dense networks were also the smallest in terms of number of nodes.

As in recent studies, we evaluate interactive visualizations. Given the different visual encoding in NL and AM it is difficult to ensure that all interactions are fair to both visualizations. To alleviate this concern we relied on a detailed review of the NL and AM literature, and selected the most common interactions and their implementations (see Section 3.3). This ensured, at least to some degree, that we evaluated the interactive visualizations as they appear in practice.

Crowdsourced studies have known inherent limitations (e.g., difficulty controlling the experimental setup and verifying what participants do). By and large, however, crowdsourcing studies replicate prior controlled lab studies [23].

6   Conclusions

We presented the results of a crowdsourced evaluation of NL and AM network visualizations. Our study involved 557 online participants who used interactive versions of the two encodings, to answer 14 varied types of questions about a large network of 256 nodes and 1090 edges. We found that NL is better than AM for questions about network topology and connectivity, and comparable for group and memorability tasks, and therefore a better choice for visualizing datasets similar to the one we evaluated, provided a similar interaction set.

References

1. Abuthawabeh, A., Beck, F., Zeckzer, D., Diehl, S.: Finding structures in multi-type code couplings with node-link and matrix visualizations. In: Software Visualization (VISSOFT), 2013 First IEEE Working Conference on. pp. 1–10. IEEE (2013)
2. Ahn, Y.Y., Ahnert, S.E., Bagrow, J.P., Barabási, A.L.: Flavor network and the principles of food pairing. Scientific reports 1 (2011)
3. Alper, B., Bach, B., Henry Riche, N., Isenberg, T., Fekete, J.D.: Weighted graph comparison techniques for brain connectivity analysis. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 483–492. ACM (2013)
4. Amar, R., Eagan, J., Stasko, J.: Low-level components of analytic activity in information visualization. In: Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on. pp. 111–117. IEEE (2005)
5. Archambault, D., Purchase, H.C., Hofeld, T.: Evaluation in the Crowd: Crowdsourcing and Human-Centred Experiments. Springer (2017)
6. Auber, D.: Tulip: A huge graph visualization framework. In: Graph Drawing Software, pp. 105–126. Springer (2004)
7. Bach, B., Pietriga, E., Fekete, J.D.: Visualizing dynamic networks with matrix cubes. In: Proceedings of the 32nd annual ACM conference on Human factors in computing systems. pp. 877–886. ACM (2014)
8. Barsky, A., Gardy, J.L., Hancock, R.E., Munzner, T.: Cerebral: a cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. Bioinformatics 23(8), 1040–1042 (2007)
9. Bastian, M., Heymann, S., Jacomy, M., et al.: Gephi: an open source software for exploring and manipulating networks. ICWSM 8, 361–362 (2009)
10. Behrisch, M., Davey, J., Fischer, F., Thonnard, O., Schreck, T., Keim, D., Kohlhammer, J.: Visual analysis of sets of heterogeneous matrices using projection-based distance functions and semantic zoom. In: Computer Graphics Forum. vol. 33, pp. 411–420. Wiley Online Library (2014)
11. Bezerianos, A., Dragicevic, P., Fekete, J.D., Bae, J., Watson, B.: Geneaquilts: A system for exploring large genealogies. IEEE Transactions on Visualization and Computer Graphics 16(6), 1073–1081 (2010)
12. Blanch, R., Dautriche, R., Bisson, G.: Dendrogramix: A hybrid tree-matrix visualization technique to support interactive exploration of dendrograms. In: Visualization Symposium (PacificVis), 2015 IEEE Pacific. pp. 31–38. IEEE (2015)
13. Borkin, M., Gajos, K., Peters, A., Mitsouras, D., Melchionna, S., Rybicki, F., Feldman, C., Pfister, H.: Evaluation of artery visualizations for heart disease diagnosis. IEEE transactions on visualization and computer graphics 17(12), 2479–2488 (2011)
14. Borkin, M., Vo, A., Bylinskii, Z., Isola, P., Sunkavalli, S., Oliva, A., Pfister, H., et al.: What makes a visualization memorable? Visualization and Computer Graphics, IEEE Transactions on 19(12), 2306–2315 (2013)
15. Chapman, P., Stapleton, G., Rodgers, P., Micallef, L., Blake, A.: Visualizing sets: an empirical comparison of diagram types. In: Diagrammatic Representation and Inference, pp. 146–160. Springer (2014)
16. Christensen, J., Bae, J.H., Watson, B., Rappa, M.: Understanding which graph depictions are best for viewers. In: International Symposium on Smart Graphics. pp. 174–177. Springer (2014)

17. Dinkla, K., Westenberg, M.A., van Wijk, J.J.: Compressed adjacency matrices: Untangling gene regulatory networks. IEEE Transactions on Visualization and Computer Graphics 18(12), 2457–2466 (2012)
18. Ellson, J., Gansner, E.R., Koutsofios, E., North, S.C., Woodhull, G.: Graphviz - open source graph drawing tools. In: Graph Drawing. pp. 483–484 (2001)
19. Elmqvist, N., Do, T.N., Goodell, H., Henry, N., Fekete, J.D.: Zame: Interactive large-scale graph visualization. In: Visualization Symposium, 2008. PacificVIS'08. IEEE Pacific. pp. 215–222. IEEE (2008)
20. Fekete, J.D.: Reorder. js: A javascript library to reorder tables and networks. In: IEEE VIS 2015 (2015)
21. Ghoniem, M., Fekete, J.D., Castagliola, P.: A comparison of the readability of graphs using node-link and matrix-based representations. In: Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on. pp. 17–24. Ieee (2004)
22. Ghoniem, M., Fekete, J.D., Castagliola, P.: On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis. Information Visualization 4(2), 114–135 (2005)
23. Heer, J., Bostock, M.: Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 203–212. ACM (2010)
24. Henry, N., Fekete, J.D.: Matrixexplorer: a dual-representation system to explore social networks. Visualization and Computer Graphics, IEEE Transactions on 12(5), 677–684 (2006)
25. Hu, Y., Gansner, E., Kobourov, S.G.: Visualizing graphs and clusters as maps. IEEE Computer Graphics and Applications 30(6), 54–66 (2010)
26. Huang, W.: Using eye tracking to investigate graph layout effects. In: Visualization, 2007. APVIS'07. 2007 6th International Asia-Pacific Symposium on. pp. 97–100. IEEE (2007)
27. Huang, W., Eades, P., Hong, S.H.: Measuring effectiveness of graph visualizations: A cognitive load perspective. Information Visualization 8(3), 139–152 (2009)
28. Jianu, R., Rusu, A., Hu, Y., Taggart, D.: How to display group information on node-link diagrams: an evaluation. Visualization and Computer Graphics, IEEE Transactions on 20(11), 1530–1541 (2014)
29. Jourdan, F., Melançon, G.: Tool for metabolic and regulatory pathways visual analysis. In: Electronic Imaging 2003. pp. 46–55. International Society for Optics and Photonics (2003)
30. Keller, R., Eckert, C.M., Clarkson, P.J.: Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? Information Visualization 5(1), 62–76 (2006)
31. Kosara, R., Ziemkiewicz, C.: Do mechanical turks dream of square pie charts? In: Proceedings of the 3rd BELIV'10 Workshop: BEyond time and errors: novel evaLuation methods for Information Visualization. pp. 63–70. ACM (2010)
32. Lee, B., Plaisant, C., Parr, C.S., Fekete, J.D., Henry, N.: Task taxonomy for graph visualization. In: Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization. pp. 1–5. ACM (2006)
33. Mason, W., Suri, S.: Conducting behavioral research on amazons mechanical turk. Behavior research methods 44(1), 1–23 (2012)
34. Melancon, G.: Just how dense are dense graphs in the real world?: a methodological note. In: Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization. pp. 1–7. ACM (2006)

35. Micallef, L., Dragicevic, P., Fekete, J.D.: Assessing the effect of visualizations on bayesian reasoning through crowdsourcing. Visualization and Computer Graphics, IEEE Transactions on 18(12), 2536–2545 (2012)
36. Okoe, M., Jianu, R.: Ecological validity in quantitative user studies–a case study in graph evaluation (2015)
37. Okoe, M., Jianu, R.: Graphunit: Evaluating interactive graph visualizations using crowdsourcing. In: Computer Graphics Forum. vol. 34, pp. 451–460. Wiley Online Library (2015)
38. Paolacci, G., Chandler, J., Ipeirotis, P.G.: Running experiments on amazon mechanical turk. Judgment and Decision making 5(5), 411–419 (2010)
39. Perin, C., Dragicevic, P., Fekete, J.D.: Revisiting bertin matrices: New interactions for crafting tabular visualizations. Visualization and Computer Graphics, IEEE Transactions on 20(12), 2082–2091 (2014)
40. Purchase, H.: Which aesthetic has the greatest effect on human understanding? In: Graph Drawing. pp. 248–261. Springer (1997)
41. Purchase, H.C., Cohen, R.F., James, M.: Validating graph drawing aesthetics. In: Graph Drawing. pp. 435–446. Springer (1996)
42. Robertson, G., Fernandez, R., Fisher, D., Lee, B., Stasko, J.: Effectiveness of animation in trend visualization. IEEE Transactions on Visualization and Computer Graphics 14(6) (2008)
43. Rodgers, P., Stapleton, G., Chapman, P.: Visualizing sets with linear diagrams. ACM Transactions on Computer-Human Interaction (TOCHI) 22(6), 27 (2015)
44. Ross, J., Irani, L., Silberman, M., Zaldivar, A., Tomlinson, B.: Who are the crowdworkers?: shifting demographics in mechanical turk. In: CHI'10 extended abstracts on Human factors in computing systems. pp. 2863–2872. ACM (2010)
45. Rufiange, S., McGuffin, M.J., Fuhrman, C.P.: Treematrix: A hybrid visualization of compound graphs. In: Computer Graphics Forum. vol. 31, pp. 89–101. Wiley Online Library (2012)
46. Saket, B., Scheidegger, C., Kobourov, S., Börner, K.: Map-based Visualizations Increase Recall Accuracy of Data. Computer Graphics Forum 34(3), 441–450 (2015)
47. Saket, B., Simonetto, P., Kobourov, S.: Group-level graph visualization taxonomy. arXiv preprint arXiv:1403.7421 (2014)
48. Saket, B., Simonetto, P., Kobourov, S., Borner, K.: Node, node-link, and node-link-group diagrams: An evaluation. Visualization and Computer Graphics, IEEE Transactions on 20(12), 2231–2240 (2014)
49. Sedlmair, M., Isenberg, P., Baur, D., Mauerer, M., Pigorsch, C., Butz, A.: Cardiogram: visual analytics for automotive engineers. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1727–1736. ACM (2011)
50. Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.: Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome research 13(11), 2498–2504 (2003)
51. Sheny, Z., Maz, K.L.: Path visualization for adjacency matrices. In: Proceedings of the 9th Joint Eurographics/IEEE VGTC conference on Visualization. pp. 83–90. Eurographics Association (2007)
52. Shneiderman, B.: The eyes have it: A task by data type taxonomy for information visualizations. In: Visual Languages, 1996. Proceedings., IEEE Symposium on. pp. 336–343. IEEE (1996)
53. Viégas, F.B., Donath, J.: Social network visualization: Can we go beyond the graph. In: Workshop on social networks, CSCW. vol. 4, pp. 6–10 (2004)

54. Von Landesberger, T., Kuijper, A., Schreck, T., Kohlhammer, J., van Wijk, J.J., Fekete, J.D., Fellner, D.W.: Visual analysis of large graphs: state-of-the-art and future research challenges. In: Computer graphics forum. vol. 30, pp. 1719–1749. Wiley Online Library (2011)
55. Ware, C., Purchase, H., Colpoys, L., McGill, M.: Cognitive measurements of graph aesthetics. Information Visualization 1(2), 103–110 (2002)
56. Yi, J.S., Elmqvist, N., Lee, S.: Timematrix: Analyzing temporal social networks using interactive matrix-based visualizations. Intl. Journal of Human–Computer Interaction 26(11-12), 1031–1051 (2010)
57. Ziemkiewicz, C., Kosara, R.: The shaping of information by visual metaphors. IEEE Transactions on Visualization and Computer Graphics 14(6) (2008)