Technical Section

Me! Me! Me! Me! A study and comparison of ego network representations

Henry Ehlers [a,*], Daniel Pahr [a], Velitchko Filipov [a], Hsiang-Yun Wu [b,a], Renata G. Raidou [a]

[a] *TU Wien, Vienna, Austria*
[b] *St. Pölten University of Applied Sciences, St. Pölten, Austria*

ARTICLE INFO

ABSTRACT

From social networks to brain connectivity, ego networks are a simple yet powerful approach to visualizing parts of a larger graph, i.e. those related to a selected focal node — the so-called "ego". While surveys and comparisons of general graph visualization approaches exist in the literature, we note (i) the many conflicting results of comparisons of adjacency matrices and node-link diagrams, thus motivating further study, as well as (ii) the absence of such systematic comparisons for ego networks specifically. In this paper, we propose the development of empirical recommendations for ego network visualization strategies. First, we survey the literature across application domains and collect examples of network visualizations to identify the most common visual encodings, namely straight-line, radial, and layered node-link diagrams, as well as adjacency matrices. These representations are then applied to a representative, intermediate-sized network and subsequently compared in a large-scale, crowd-sourced user study in a mixed-methods analysis setup to investigate their impact on both user experience and performance. Within the limits of this study, and contrary to previous comparative investigations of adjacency matrices and node-link diagrams (outside of ego networks specifically), *participants performed systematically worse when using adjacency matrices than those using node-link diagrammatic representations*. Similar to previous comparisons of different node-link diagrams, *we do not detect any notable differences in participant performance between the three node-link diagrams*. Lastly, our quantitative and qualitative results indicate that *participants found adjacency matrices harder to learn, use, and understand than node-link diagrams*. We conclude that in terms of both participant experience and performance, a layered node-link diagrammatic representation appears to be the most preferable for ego network visualization purposes.

1. Introduction

Ego networks, sometimes also called egocentric or personal networks [1], are node-relative subgraph depictions of a larger graph's topology (Fig. 1). That is to say, instead of drawing a graph in its entirety, one only draws those nodes and edges relevant to some selected focal node. This focal node is commonly referred to as the "*ego*", and its neighbors as "*alters*". More specifically, the ego's immediate neighbors, i.e. nodes of a 1-hop distance from the ego, are called 1-alters. The ego's neighbors' neighbors, i.e. nodes a 2-hop distance from the ego, are called 2-alters, and so on. This subgraph of a selected ego node and its *k*-alters is called an ego network.

Ego network visualizations, especially widespread in the social sciences [2,3], are a conceptually simple, yet powerful approach for interactively reducing the visual complexity of larger networks. Consider here a number of such examples: Pu et al. [4] go beyond traditional supervised and unsupervised methods and employ ego network visualizations to enhance anomaly detection for fraud identification.

Liu et al. [5] used split ego network representations to enable social scientists to tackle the challenging task of comparing two ego networks simultaneously. However, ego network representations of graphs have found (admittedly sparse) application outside of social network analysis as well. For one, Al-Awami et al. [6]'s *NeuroLines* framework visualizes nanoscale neuronal connectivity as node-relative networks. Alternatively, Sayers et al. [7] proposed a node-centric visualization of Resource Description Framework (RDF) graph topology.

In many applications, relationships within an ego network carry nuanced meaning that depends heavily on their importance. For instance, in social networks, the importance of relationships is represented as a weight (i.e., strength or frequency of interactions), allowing for analysis of close versus casual associations [2]. Similarly, in neuroscientific studies, weighted connections capture the intensity or frequency of neural interactions for identifying dominant pathways [6]. Weighted ego networks support tasks to reveal specific association

---

* Corresponding author.
E-mail address: henry.ehlers@tuwien.ac.at (H. Ehlers).

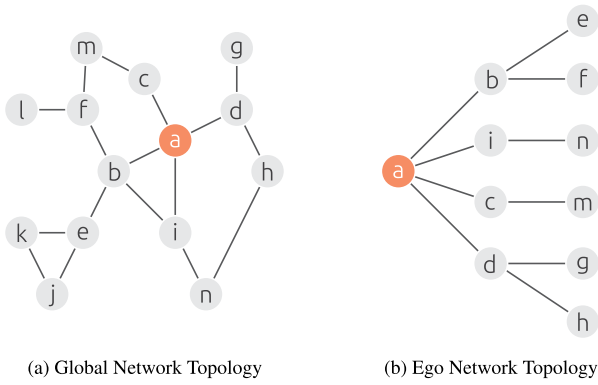(a) Global Network Topology   (b) Ego Network Topology

Fig. 1. Construction of a 2-alter ego network representation (b) from some given input graph (a). The selected ego, i.e. node **a**, is highlighted in orange in both the network's global topological representation as well as its own ego network. The alters of the ego are shown in a tree-like structure, i.e. all its 1-alters are depicted on the tree's first layer, and all its 2-alters on the second.

patterns (i.e. identifying the most influential or strongest connections).

Depending on the particular application domain and graph data at hand, different types of network embeddings are employed—from conventional straight-line node-link diagrams [2] to dynamic adjacency matrices [8]. However, *no systematic work has yet been conducted to study ego networks, their representations, or their applications*. Looking at general network visualization, there are several, still very much conflicting, user performance and preference studies comparing different types of network representations. For instance, node-link diagrams have been compared to adjacency matrices [9,10] or different types of node-link diagrams to each other [11,12], in order to determine what types of tasks and data are best served by which representations.

Ego networks are graphs centered around some particular node of interest, the ego, and are typically smaller-scale networks, compared to conventional non-ego networks. Moreover, ego networks, typically, are employed for very different reasons than non-ego networks. Thus, while certain findings of such general network visualization comparisons transfer partially to ego networks and their visualization, their particular analytical goals, analytical tasks, and challenges warrant isolated and focused investigation. Here, we anticipate conventional straight-line node-link diagrams to be outperformed by more complex and structured visualizations that can better represent the layered nature of ego networks — even if such visualizations prove more difficult to learn and understand. To the best of our knowledge, these issues have caused visualization researchers to lack guidelines and best practices on when and how to use ego network visualizations effectively; a gap we aim to start filling with this paper.

More specifically, we aim to highlight ego networks, commonly pigeonholed as a social network analysis-specific approach, as a potentially effective network visualization technique across domains that deserves greater attention. We make a first step in this direction by identifying which representations are common in the field and understanding how effective they are for particular graph analysis tasks. To do so, we provide an overview of various (weighted and) undirected ego-network applications, tools, and examples across different domains through an extensive survey of the literature of 50 papers. With the survey's results in hand, we identify four common approaches to visualizing ego networks, namely *(i) straight-line node-link diagrams* (✕), *(ii) layered (tree-like) node-link diagrams* (▲), *(iii) radial node-link diagrams* (●), *and (iv) adjacency matrices* (▧). To identify when a specific representation is preferred over another, we conduct a large-scale crowd-sourced user study of 120 participants to empirically investigate the effect the layouts have on user performance and experience across six ego network-specific low-level graph analysis tasks. Additionally,

we probe perception and preference in order to gauge which representations are easiest to learn and use. Finally, to go beyond a purely quantitative evaluation of user performance or preference, we collect written user feedback throughout the conducted study whose qualitative analysis is used to contextualize and understand more deeply our statistical results.

In summary, the contributions of this paper are:

1. an overview of ego network representations across application domains (Section 3),
2. a large-scale user study to investigate, both quantitatively and qualitatively, the effect of these representations on user performance and preference (Section 4), and
3. a discussion of the (conceptual) (dis)advantages of the four selected ego network representations to provide recommendations on their usage (Section 6).

2. Related work

As we are interested in studying and comparing different layout approaches to ego network visualization, we deem it important to understand what systematic evaluations have been conducted both outside of and within the context of ego networks. Specifically, in this section, we discuss comparison studies between adjacency matrices and node-link diagrams, user performance and graph aesthetics of different node-link diagrammatic layouts, and various evaluations conducted within the context of ego network visualization.

2.1. Comparisons of matrices and node-link diagrams

In their seminal study, Ghoniem et al. [9,13] compare the "*readability*" of node-link diagrams and adjacency matrix representations: the more readable the graph, the faster and more accurately a participant can complete a series of low-level graph tasks. With the results of a completed user study, the authors conclude that node-link diagrammatic representations are more suited for smaller and sparser graphs, whereas matrix representations should be preferred for larger and denser ones—with the possible exception of path-tracing tasks which proved difficult in both representations. Several follow-up works have further studied the differences between node-link diagrams and adjacency matrices across graph analysis tasks [14,15], as references against some third novel hybrid representation [16–18], and across application domains, from brain connectivity networks [19] to terror networks [20]. Here, we discuss the results of those follow-up studies for different graph tasks.

Adjacency tasks deal with the identification of a given node's immediate neighborhood, i.e. listing or counting its neighbors. Results differ depending on the size of the network under study. For larger networks, node-link diagrams appear to be favorable [21,22], whereas for smaller and medium-sized graphs results are less conclusive, with some finding adjacency matrices to be favorable [16,23], other finding node-link diagrams to be preferable [10,24], while others find no difference between the two representations at all [25,26]. **Accessibility** tasks, closely related to adjacency tasks, concern themselves with the identification of incident edges between given nodes, e.g. checking whether two given nodes are indeed adjacent to each other. Contrary to Ghoniem et al.'s [9,13] (non-statistically significant) initial findings, subsequent work, across graph sizes, appears to point towards the superiority of node-link diagrams over matrices [21,24,25,27,28]. Indeed, only one paper, a replication of Ghoniem et al.'s original study, showcases the statistical superiority of adjacency matrices [29]. **Common connection** tasks deal with the identification of nodes adjacent to not just one, but two or more given nodes, e.g. locating the common neighbors of a set of given nodes. In their original work, Ghoniem et al. [9,13] were unable to showcase any statistical differences between the two representations. Here, results tentatively

favor node-link diagrams: several works have shown the superiority of node-link diagrams for the identification of common connections [16, 24–27], while others have shown the opposite [19,21]. Many were unable to find any statistically significant results [10,22,30]. **(Shortest) path-finding** tasks challenge the user with tracing multiple possible paths between two given nodes and reporting the shortest one. Here, Ghoniem et al.'s [9,13] initial findings, i.e. the superiority of node-link diagrams over adjacency matrices, has been confirmed by effectively all follow-up studies [10,21,22,24]. Relatedly, **path-following tasks** have also been systematically shown to be more accurate using node-link diagrams [10,21,22]. Lastly, **overview** tasks deal, very generally, with gaining a big-picture understanding of a graph, such as identifying a graph's class [31] or estimating the graph's density [32]. In this case, while Ghoniem et al. [9,13] found adjacency matrices to be superior across all overview tasks, results since have been mixed, with some finding node-link diagrams to be superior [32], others adjacency matrices [31].

In general, looking at the last twenty years of comparisons, not all of Ghoniem et al.'s [9,13] initial findings can be taken as fact. The collected 21 comparisons certainly highlight the superiority of node-link diagrams over matrix representations for path-finding, path-tracing, and accessibility tasks, both in terms of task accuracy and completion time. For topology or attribute-related tasks, it is difficult to draw any similarly sweeping conclusion. Still, for increasingly large and dense graphs, performance in node-link diagrams does appear to be more heavily affected than in matrices [21,32].

2.2. Comparisons of node-link diagram layouts

Node-link diagrams can take many different shapes and forms depending on the particular layout approach. Classical node-link diagrams can be drawn using a wide variety of force-directed algorithms and spring embedders. Groups within these graphs can be highlighted by embedding them using group-based layout techniques or laid out radially or hierarchically to highlight other structures within the data. However, it is unclear for which situations certain graph layouts are to be preferred.

Early evaluations, such as the work of Blythe et al. [33] and Purchase et al. [34] went beyond **comparing classical layout algorithms** in terms of their produced graph aesthetic criteria and focused their efforts on studying their effect on human performance. More recently, Meulemans and Schulz [12] studied the effect of three different graph layout algorithms within the context of social networks on human perception. In general, while (statistically significant) differences between layouts can be detected, no meaningful differences between individual pairwise layouts can be identified (despite these layouts differing substantially in terms of their graph aesthetic metrics). **Beyond comparisons of classical** node-link diagram layout algorithms, research has focused on comparing, for example, different approaches to "sociogram" layouts, i.e. radial, hierarchical, group, and free layouts [11], or tree representations to classical node-link diagrams [35]. Once again, no meaningful conclusion regarding the impact of graph layout on performance can be drawn. Lastly, Didimo et al. [24] compared four different approaches to the directed graph layout problem, namely, hierarchical, orthogonal, overloaded orthogonal layouts, and matrix representations. The results, however, indicated that, in terms of error rates (for the particular data studied) overloaded orthogonal drawings outperformed all other layout approaches.

In general, depending on the particular dataset presented, node-link diagram layouts indeed seem to have an impact on human perception, i.e. different layouts result in different task accuracies and completion times. However, it is unclear whether a single "best" layout can be identified, as pairwise statistical differences between layouts are often inconclusive. It is safe to conclude that any differences between such layouts are either fairly minimal or highly task/data-dependent.

2.3. Ego-network evaluations

Most commonly, ego network visualization evaluations have been conducted either as usage scenarios [36], or as case studies [6,37,38]. In some instances, additional early [2] or final user feedback [38–40] was also collected. However, task-based evaluations are featured in several papers. For example, within the context of **dynamic ego network visualization**, e.g. visualizing the evolution of such networks over time, several task-based evaluations have been conducted [41,42]. These evaluations naturally deal predominantly with time-dependent tasks, such as summarizing the evolutionary trend of clusters for a particular ego [3], identifying topological changes in the ego's connectivity [43], or determining whether the 1-alter subgraph increased or decreased in size over time [8]. Alternatively, within the context of (dynamic) **comparative ego network visualization approaches**, such as Liu et al.'s *EgoComp* [5] or Wu et al.'s *EgoSlider* [1], similar task-based evaluations can be found. In such cases, tasks focus on evaluating differences between egos, such as identifying which ego has the largest number of 1-alters with certain properties, characterizing the overall similarity between two selected egos, or inspecting whether a particular alter of one ego exists in the network of another. Most importantly, **non-dynamic task-based evaluations focusing on a single ego** can also be found in Shikora et al.'s *InfluViz* [44] and Sorger et al. [45]'s virtual reality ego network visualization. In such cases, participants completed multiple topological tasks derived from Lee et al.'s [14] graph analysis taxonomy, such as finding common neighbors of two alter nodes, estimating the degree of a particular given node, or finding a (shortest) path between two given nodes. Unlike previous studies that have primarily focused on either quantitative or qualitative evaluations of ego networks, our study introduces a novel mixed-methods approach that integrates both methodologies. This allows us to provide a more comprehensive examination of differences between various ego network visualization techniques and a more nuanced understanding of how different representations impact the interpretation of ego network structures, compared to prior research.

3. Ego network representations

To better understand the current state of ego network visualization, we perform a literature survey, collecting 50 papers, comprising visualization techniques, systems, and application papers that feature ego network visualization approaches. A complication to this search is the lack of consensus on terminology to describe such networks. Examples include ego networks, egocentric networks, node-relative networks, or subject-relative networks. In other cases, such networks are not explicitly identified as ego networks at all [6]. Owing to these difficulties, the literature survey was conducted manually, driven by keyword searches across multiple academic search engines, and, in a snowballing method, combining through their bibliographies exhaustively. These collected works were then manually read and filtered to ensure relevance to the project. Each of the thus collected 50 references is categorized based on the visual representation featured as well as the domain's application area (see Fig. 2). Ultimately, five common graph representations are identified, namely (i) straight-line node-link diagrams (✕), (ii) radial node-link diagrams (🔴), (iii) trees/layered node-link diagrams (🔺), (iv) adjacency matrix representations (🟪), and (v) latent variable space embeddings. However, as latent variable embeddings commonly omit drawing edges altogether, they are unsuitable for topology-based tasks that we investigate (Section 4.2.2). Therefore, we omit them from both this section as well as our user study. Given the large prevalence of weighted ego networks in literature, particularly in applications requiring precise association analysis, we focus on weighted representations for this study. Weighted networks offer critical insights into the strength and frequency of connections, which are pivotal in real-world tasks across diverse domains (see Section 2).
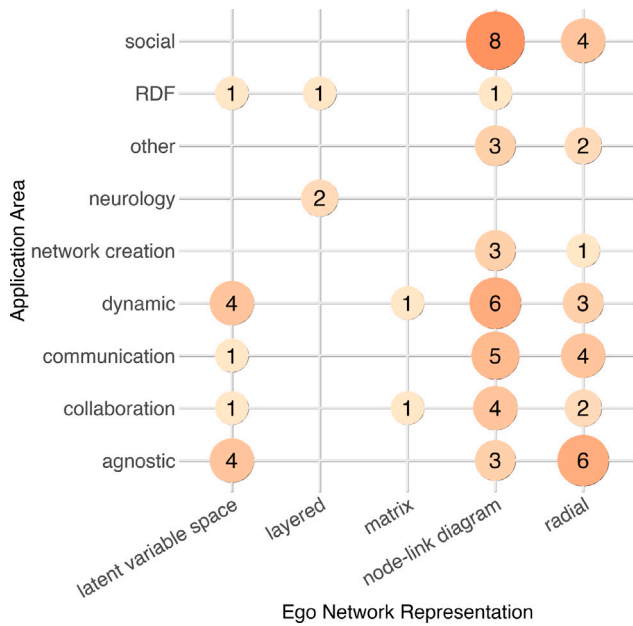
Fig. 2. Mapping of the 50 collected ego network papers to both application area and ego network representation. The number of papers that map to a combination of categories is encoded in its circle's size, color intensity, and numerical label. Papers could map to multiple representations and application areas.

In this section, we briefly discuss the four remaining archetypal ego network representations that are commonly used as well as how they were designed and implemented. The implementation, as well as our classification of the collected papers, has been made available on the Open Science Framework.[1]

3.1. The straight-line node-link diagram

Straight-line node-link diagrams (✕) are arguably the most well-known graph representation for networks. For some given graph $G = (V, E)$, its nodes V are represented as points, circles, or rectangles placed freely in 2D space connected by straight line segments, representing its edges E [46] (see Fig. 3(a)). Such diagrams also proved to be the most popular form of ego network visualization, featured in 28 unique papers. In its simplest form, such ego-centric node-link diagrammatic representations lay out the graph's global topology and then (interactively) highlight egos and their 1-alters, as presented in Fisher et al.'s [47] visualization of communication networks. In other cases, the ego and its incident edges are not explicitly visualized at all, and, instead, only the ego's 1-alters and their intra-1-alter connectivity are visualized, as discussed in the work of Ezaiza et al. [2], which focuses on various friendship clusters in personal social networks. However, we note that such representations naturally do not allow users to investigate the ego–alters' edge weights. Here, edge weights were encoded in each edge's line segment's opacity, i.e. the greater the edge weight, the more opaque the edge. This ensures that edges, much like their node counterparts, remain the same size/thickness both within and across node-link diagrammatic representations. Lastly, opacity is also chosen to ensure comparability to the to-be-discussed layered adjacency matrix representation.

3.2. The radial node-link diagram

A radial node-link diagrammatic representation (🔴) is a more constrained form of the previously discussed straight-line node-link di-

agrams. Instead of allowing nodes to be drawn freely in 2D space, their placement is restricted to a given circle. Edges are then drawn either as straight line segments or smooth arcs, mostly within the confines of the given circle. With 19 papers, radial node-link diagrams form the second most common representation of ego networks. In their simplest form, such radial representations display the ego at the center of the circle and its 1-alters radially around it, with both ego–alter and alter–alter edges displayed within the formed circle's area (see Fig. 3(b)) [36]. As discussed previously, ego–alter edges are sometimes omitted and only intra-1-alter connectivity is visualized [2]. Multiple alter-levels can also be visualized as multiple concentric circles, i.e. 1-alters placed along the first ring, 2-alters along the second ring, and so on. In such representations, intra-alter edges are commonly omitted and only inter-alter edges rendered [3,42]. However, these concentric circles need not only denote different alter levels. For example, in the context of dynamic ego network visualization, each ring can indicate a particular time slice instead. An ego's various 1-alters' presences/absences are then indicated for each such time slice [48]. Lastly, for the sake of completeness, radial layouts are also utilized for the comparison of egos' 1-alters by, for example, juxtaposing their concentric alter rings and rendering ego–alter edges not only within each circle but also between them [49]. Again, edge weights are encoded using each edge's line segment's opacity.

3.3. The layered node-link diagram

Similar to radial node-link diagrams, layered node-link diagrams (▲) opt to restrict node placement instead of allowing them to be drawn freely in 2D space. Unlike radial diagrams, such representations arrange nodes not along the circumference of concentric circles, but instead along equidistant lines called layers. Edges are commonly represented as straight-line segments between them. For ego networks specifically, layered node-link diagrams are tree-like representations of ego networks, which are notably less popular than straight-line or radial node-link diagrams, with only 3 papers mapping to this particular category. Here, the aforementioned layers in layered ego network node-link diagrams encode the specific alter level, and nodes are placed along it accordingly (see Fig. 3(c)). Consider, Sayers et al.'s [7] RDF graph visualization, which provides a very representative example hereof. While in all examples found, only inter-alter edges were rendered, we opt to render all edges of the ego network to investigate their utility for both inter and intra-alter connectivity analysis. Edge weights are again encoded using each edge's line segment's opacity.

3.4. The layered adjacency matrix

Unlike the previously discussed node-link diagrams, an adjacency matrix representation (🟪) takes the form of a data table, in which nodes are represented both as rows and columns. An undirected edge connecting two particular nodes is then represented by "filling" the two corresponding matrix cells of the symmetric table with a 1 (0 otherwise). For example, should nodes $v_i, v_j \in V$ be connected by some undirected edge $\{v_i, v_j\} \in E$, the matrix cells (i, j) and (j, i) are correspondingly "filled in". Here, in line with the conclusions of previous surveys of network visualization outside of the context of ego network visualization, only very few approaches make use of (adjacency) matrix representations [50,51]. Across the 50 collected ego network visualization papers, only one uses an adjacency matrix representation, namely Zhao et al.'s dynamic egocentric network representation *EgoLines* [8].

In order to meaningfully exploit and visually display an ego network's topological connectivity as well as its various alter levels, and ensure it is comparable to both radial and layered node-link diagrams, we visualize these networks as *layered* adjacency matrices (see Fig. 3(d)), akin to a centered matrix representation [52] or a quilt [53, 54]. In such representations, nodes are grouped by some layer structure (here their k-alter level). This ensures that all intra-k-alter connectivity

(a) Straight-Line Node-Link Diagram   (b) Radial Node-Link Diagram   (c) Layered Node-Link Diagram   (d) Layered Adjacency Matrix
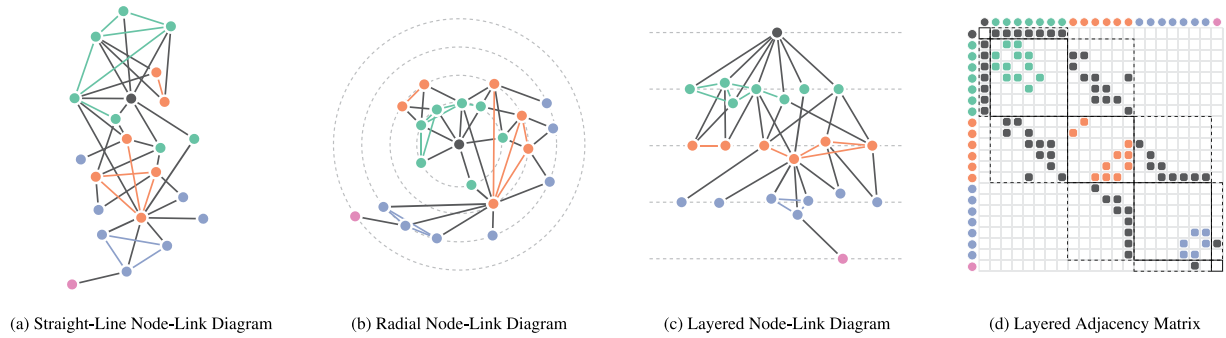
Fig. 3. Illustrative visualizations of the implemented ego network representations. Nodes and intra-alter edges are colored-coded according to their alter-level using an appropriate color palette. More specifically, 1-alters are colored green, 2-alters orange, 3-alters blue, and 4-alters purple. Inter-alter edges, as well as the ego, are colored dark gray.

is displayed only within a block of k-alter nodes. Additionally, because a k-alter can, by definition, only be connected to $(k-1)$ or $(k+1)$ alters, all inter-alter connectivity is displayed within blocks adjacent to their corresponding intra-alter blocks. This, in turn, results in larger portions of the off-diagonal elements remaining empty, resulting in overall less visual clutter. To provide a meaningful one-dimensional ordering of the nodes (an important consideration when evaluating adjacency matrix representations [22,24,25]), nodes are sorted according to their graph-theoretic distance to the ego within each alter level. Hop-distance ties within a block of k-alters were broken using their weighted distance to the ego, i.e. the closer their graph-theoretic distance, the closer their visual distance to the ego within their corresponding k-alter block. In line with the previously discussed node-link diagrammatic representations' visualization of edge weight, we opt to encode edge weight using opacity. More specifically, the greater the edge weight of some undirected edge between nodes v_i and v_j, the more opaque the corresponding matrix cells (i, j) and (j, i).

3.5. Implementation

For our implementations, many different spring-embedded and force-directed approaches to laying networks as node-link diagrams present themselves [55]. For the straight-line, radial, and layered node-link representations, we make use of *D3js*'s [56] particle-based force-directed algorithm to lay out the networks, as it produces consistently visually pleasing results while remaining computationally tractable for the kinds of undirected, weighted graphs we consider here. It is important to note that while the four node-link diagrammatic representations may, in certain circumstances, share visual similarities, they should never look identical to each other, thereby distinguishing themselves from each other both conceptually and visually. The layered adjacency matrix was also implemented in *D3js*. Additionally, across all representations, in order to visually communicate the alter levels, nodes, and intra-alter edges are color-coded using an appropriate *Color Brewer* [57] palette (see Fig. 3). Specifically, 1-alters are colored green, 2-alters orange, 3-alters blue, and 4-alters purple. Inter-alter edges, as well as the ego, are colored dark gray. Similar to previous (ego) network visualization evaluations, we also implemented basic interactivity [10,22,25]. Interactions include highlighting (in red) incident edges and adjacent nodes when hovering over a node for all node-link representations. For the adjacency matrix, when hovering over a cell (edge) at location (i, j), row i, column j, and nodes v_i, $v_j \in V$ are also highlighted in red. Additionally, basic navigation, i.e. panning and zooming, are provided. We refer the reader to the supplement's Figure 1 or the previously mentioned OSF repository for visual examples of the implementation as presented to study participants.

4. Study

We aim to evaluate the ego network representation's effect on user *performance* and *experience* in a mixed methods approach in order to better understand and contextualize our quantitative results. Given that ego networks are highly specialized tools, a group of expert users would be optimal to test these effects, such as network scientists or domain experts, e.g. social scientists or biologists. However, to achieve adequate statistical power and sufficient numbers of qualitative feedback, we instead opt to conduct an online, crowd-sourced user study.

First, we wish to quantitatively evaluate the effect of ego network representation on participants' performance, i.e. their ability to complete a series of low-level graph analysis tasks as quickly (response times) and correctly (accuracy) as possible. To do so, we employ a between-subjects study design, in which each participant completes six low-level graph analysis tasks for one randomly assigned archetypal representation. Here, a between-subjects study design was selected to ensure the study could be completed in less than 30 min in an online setting, which (based on our prior experiences with online studies) should be considered the upper limit for such online studies. Requiring each participant to complete all six tasks for all four representations would have (i) introduced a possible learning effect, and (ii) taken too long, thereby affecting participant concentration. To curb any further systematic learning effects in our results, the order in which participants are asked to complete these six tasks is randomized, and the presented graph's nodes' labels are randomized for each task, mitigating memorization.

Second, we aim to quantitatively evaluate the effect of these representations on user experience. To assess this aspect, each participant is presented with five rankable statements, relating to the ease of use and learning, participants' perceived accuracy and efficiency, as well as the aesthetic appeal of the presented visualization, which they are required to answer on a 5-point Likert scale.

Third, we aim to enrich these quantitative analyses with additional qualitative data. So, after each of the tasks given to a user, we present the task's description again and ask the users to provide a short comment on how the assigned ego network representation assisted or hindered them in the task performed. Finally, at the end of the survey we collect participant feedback pertaining to their final thoughts about the assigned ego network representation.

4.1. Graph data

Similar to the evaluation conducted by Okoe et al. [21], we opt to investigate a single network; more specifically, the real-world network representing the "*Les Misérables*" character interaction graph, consisting of $|V| = 77$ nodes and a total of $|E| = 254$ undirected edges, in which
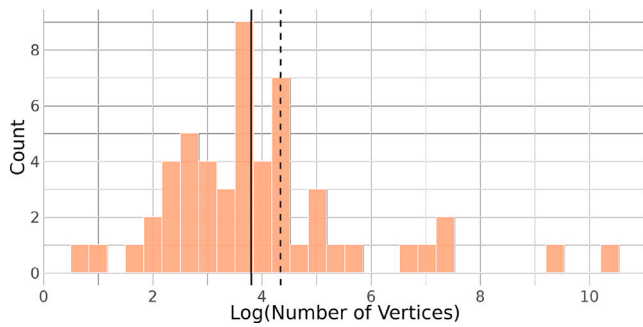
Fig. 4. Studied network sizes of 53 ego-networks from our gathered 50 papers. For several papers, no specific sizes were specified nor could have been estimated from figures. Network size is displayed on the log scale for readability. The overall median network size, $|V| = 45$, is illustrated by the dashed black line, and the chosen "*Les Misérables*" character interaction graph, $|V| = 77$, by the dashed black line [58].

edge weights encode the number of times two particular characters co-occurred [58]. To simulate the process of creating an ego network from a larger network, with "*Valjean*" as the selected ego, an ego network is constructed from this graph by computing the shortest hop-distance paths from the ego to each other node. Ties in path length are broken by their weighted distances. Labels were randomly generated integers to ensure that, even though a user was investigating the same network, the network's labels were never the same across the various tasks investigated (Section 4.2.2). Here, this particular network was chosen for six key reasons. First, by utilizing a real-world network, we avoid non-representativeness often associated with simulated (scale-free) graph data [59]. Second, by focusing on a single network, we are able to statistically investigate a larger number of tasks relevant to ego networks without sacrificing statistical power. Third, depending on the particular ego chosen, here "*Valjean*", said dataset features a larger number of alter levels than most investigated data while remaining overall representative: while most (31) ego networks focused on either 1 or 2-alters, several (7) also studied ego networks of 3-alters and higher. Fourth, by evaluating the size of ego networks investigated in previous approaches and applications, we determined this particular dataset's size to be slightly larger than the median graph size studied in ego network literature while remaining representative (see Fig. 4). Moreover, a graph of this size and density was deemed appropriate for the non-expert user group that would be tasked with analyzing it. Fifth, in order to build on work done previously, this network has already found use as a benchmark in both ego network [5] as well as more general network visualization evaluations [24]. Lastly, with $|V| = 77$ nodes and a total of $|E| = 254$ undirected edges, this particular network falls nicely within the most commonly explored/visualized sizes of graphs [60], thereby further ensuring it is representative in terms of its numbers of nodes and edges.

4.2. Procedure

To conduct our mixed methods analysis of user experience and performance, we must ensure that participants are properly instructed and trained before they commence a series of graph analysis tasks and a final user experience evaluation.

4.2.1. Training

In line with other comparative network visualization studies, we employ pre-study training to familiarize participants with the upcoming representations and tasks. Following the definitions of Nobre et al. [10], we use a mixture of active and passive training. Each participant was presented with both (i) a written tutorial explaining the details of ego networks and the assigned network representation to familiarize them with concepts and terminology, as well as (ii) an interactive

visualization of a simple network using the network representation they could expect during the evaluation to familiarize themselves with the modes of interaction we provide. Key definitions remained available to participants in a small glossary throughout the study to avoid terminology-related errors.

4.2.2. User performance: Tasks

Following existing quantitative (ego) network evaluations (Section 2.3), we are interested in statistically comparing how quickly and accurately participants were able to complete a series of low-level graph analysis tasks using the different representations we provide (see Section 3). However, a key challenge here lies in the selection of a set of tasks: they must remain general enough for their results to transfer to other network types, yet be specific enough to be meaningful for ego networks specifically. We derive our set of tasks from existing ego network evaluations. Of the 50 collected papers, nine featured a quantitative, task-based evaluation of their proposed techniques.

From these evaluations, we identify twelve common tasks when placed within Lee et al.'s [14] low-level graph task taxonomy: one browsing task, two overview tasks, and nine topological tasks (Fig. 5). More specifically, these twelve tasks include (i) following the path between a series of nodes, (ii) estimating the number of (k-) alters, (iii) estimating the number of unique edges among alters or between 1-alters and ego, (iv) identifying the alter closest to the ego, (v) identifying the node (ego or alter) with the highest adjacency, (vi) identifying the shortest path between two nodes, (vii) identifying the alter closest to some other alter, (viii) identifying the common neighbors of two given nodes, (ix) counting the neighbors of a given alter, (x) identifying the neighbors of a given alter, (xi) determining whether an edge between two nodes exists, and (xii) finding bridges between clusters.

It should also be noted that several of these dynamic egocentric network visualization system evaluations [1,3,43,62] also featured extensive time-dependent tasks, such as estimating changes of alter numbers between time steps [43] or estimating the number of relationships that lasted for a certain number of time steps [8]. However, as the scope of this paper is explicitly related to investigating differences in the perception and execution of topological ego-centric tasks across different ego network representations, we exclude tasks that focus on either the temporal or attribute-based nature of data. We argue that before one investigates tasks related to more complex multivariate or dynamic graph data, potentially requiring more sophisticated interaction techniques and training, it is necessary to first understand the effect of ego network representation in the context of relatively simpler and mostly static graphs. Therefore, from the thirteen common tasks, we selected the six most common to investigate in our user performance study [63], i.e., two overview tasks, two topological adjacency tasks, one topological common neighbor task, and one topological association task.

Overview tasks [14] can take many different shapes and forms, such as the identification of a network's graph class [31,32], the counting or approximation of a network's nodes and/or edges [9,13], or the estimation of a network edge density [32]. Based on the ego network-specific overview tasks we identified in the literature, we investigated two such tasks:

T_{edges} Count the intra-alter edges between all 1-alters.

T_{alters} Count the number of 2-alters in the graph drawing.

Adjacency tasks focus on the immediate adjacency of a node, i.e., its neighbors, and can take several forms, such as counting a given/highlighted node's neighbors [29] or edges [26], counting incoming and outgoing edges separately [32], or finding the most connected node [9,13], the highest degree node [23], or the node with the highest number of edges [16]. Based on our literature review, we investigated tasks:

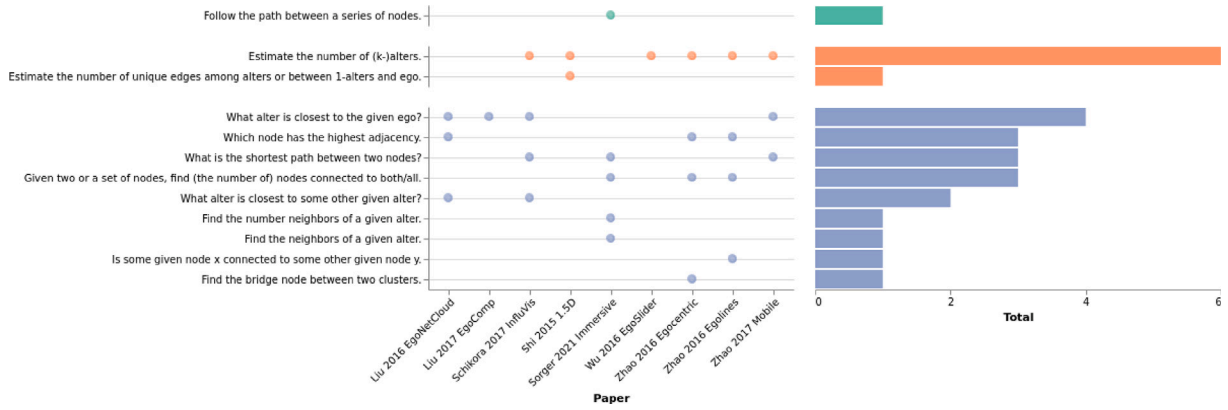$T_{neighbors}$ Count the neighbors of a given alter.

Fig. 5. Categorization of (non-dynamic and attribute-related) graph analysis tasks according to nine previous works that featured a task-based evaluation of ego networks (i.e. Liu et al. [42], Liu et al. [5], Shikora et al. [44], Shi et al. [41], Sorger et al. [45], Wu et al. [1], Zhao et al. [8], Zhao et al. [8], and Zhao et al. [61]). Additionally, the twelve tasks are categorized, faceted, and color-coded according to Lee et al.'s [14] graph task taxonomy, i.e. Browsing Tasks, Overview-based Tasks, and Topology-based Tasks.

T_{degree} Which 2-alter has the highest degree of all 2-alters?

Common neighbor tasks involve the identification of two (or more) nodes' common neighbors and subsequently list them by their identifier [16,25] or count them [26]. Here, we specifically investigated the former:

T_{common} Which neighbors do two alters have in common?

Finally, **accessibility** tasks [14] involve the identification of incident edges between nodes, such as finding the edge between two given nodes [9,13], determining whether a node is connected to another in two hops or fewer [25], or determining whether two given nodes are connected or not [21]. Based on our literature review, we specifically tasked participants with:

$T_{association}$ Which alter's edge to the ego has the largest weight?

Note that for tasks T_{degree} and T_{alters}, $k = 2$ was chosen to ensure there was a layer of alters, i.e. $k = 1$ and $k = 3$, surrounding it to better capture the visual complexity of arbitrary-sized ego networks. Moreover, it should be noted that despite its frequent occurrence in ego network evaluations (Fig. 5), we opted to ignore shortest path identification tasks owing to the by now well-documented superiority of node-link diagrams over adjacency matrix representations [10,21,22, 24] which may bias our results. Additionally, as there always exists a path between two nodes through the ego, therefore, the (shortest) path lookup becomes a trivial task.

4.2.3. User experience: A Likert scale

In order to probe each participant's perceived and subjective effectiveness and preferences [63], at the end of the study, similar to existing ego network evaluations [3,8], each participant is required to answer five statements on a 5-point Likert scale:

S_{learn} I found the ego network's visual representation easy to learn.

S_{use} I found the ego network's visual representation easy to use.

$S_{pleasing}$ I found the ego network's visual representation aesthetically pleasing.

$S_{accurately}$ I found the ego network's visual representation allowed me to answer questions accurately.

$S_{quickly}$ I found the ego network's visual representation allowed me to answer questions quickly.

4.2.4. Qualitative feedback: Participant comments

Finally, to go beyond a purely quantitative evaluation, we additionally probe both user performance and experience qualitatively. To

do so, participants were required to provide written feedback after each completed task as well as at the very end of the survey together with their Likert-scale-ranked experience. Specifically, we asked for feedback on how the representation was helping with or hindering each task's completion and the participant's general experience with the representation.

4.3. Analysis

The collected data must be evaluated both statistically and qualitatively. To do so, we formulate a series of hypotheses regarding our quantitatively evaluated metrics, i.e. task accuracy, task completion time, and user preference. Subsequently, we outline how the statistical analysis is conducted. Finally, we outline how the coding of the qualitative analysis is performed.

4.3.1. Hypotheses

Based on comparisons of node-link diagrams and adjacency matrices (Section 2.1), as well previous comparisons of ego networks (Section 2.3), we formulate four hypotheses regarding user performance and experience.

H_1: Counting on matrices. The overall findings in the literature seem to tentatively point towards the superiority of adjacency matrix representations over node-link diagrams for non-path-tracing/finding and accessibility tasks, especially for larger and denser graphs [13,32]. Subsequently, though we do not study the densest and largest graphs possible to not overwhelm non-expert users, we hypothesize that our layered adjacency matrix representation should result in higher accuracy than the three node-link diagrammatic representations for such tasks, i.e. (i) counting the number of edges (T_{edges}) and 2-alters (T_{alters}), (ii) estimating the 2-alter with the highest degree (T_{degree}), as well as (iii) counting the number of neighbors of a given alter ($T_{neighbors}$). However, given past results comparing node-link diagrams and adjacency matrices [10,25,64], we also hypothesize this increased accuracy will require additional time.

H_2: Node-link nuance. While past evaluations comparing different types of node-link diagrams have not shown meaningful differences [12,33,65], we expect the inherently layered structure of ego networks to tease out differences between the three node-link diagrammatic representations for the investigated overview and adjacency tasks. Specifically, given the 1D arrangement of nodes in both radial and layered node-link diagrams, we anticipate said representations to outperform the classical node-link diagram for the 2-alter-counting task (T_{alters}). For layered node-link diagrams especially, the clarity of this 1D arrangement of nodes comes at the cost of effectively visualizing intra-alter edges. Subsequently, we anticipate the layered node-link diagram

to perform worse than the radial or classical node-link diagrams for the intra-alter edge counting task (T_{edges}). Lastly, while disadvantageous for node/edge counting tasks, the unbounded embedding of the classical node-link diagram should communicate the neighborhood of individual nodes more effectively. As a result, we anticipate this representation to lead to more accurate results than the radial or layered node-link diagrams for the two tasks related to node neighborhood, i.e. T_{degree} and $T_{neighbors}$.

H_3: Topologically topsy-turvy. Given the findings of individual papers pointing tentatively towards the superiority of node-link diagrams for certain topological tasks [13,25], we anticipate the three node-link diagrammatic representations to outperform the layered adjacency matrix for the two topological tasks under study, i.e. locating the ego's most closely associated alter ($T_{association}$) and identifying the common neighbors of two given alters (T_{common}). Given this difficulty, we additionally hypothesize the matrix representation will require participants to spend more time answering these tasks. Within these three node-link diagrammatic representations, however, we anticipate the straight-line node-link diagram to outperform the other two for task $T_{association}$ in particular, as closely associated nodes can be placed closer to each other in 2D space, whereas layered and radial representation limit node placement along layered lines or concentric circles, respectively.

H_4: Where the instruction booklet at. Looking at past studies comparing node-link diagrams and adjacency matrix representations, it would appear as though (i) node-link diagrams are easier to learn than adjacency matrices [10,25], and (ii) participants might be more familiar with node-link diagrammatic representations *a priori* [10,32]. Subsequently, we hypothesize that participants will find the three different types of node-link diagrams easier to learn (S_{Learn} and S_{Use}) than the adjacency matrix representation. Additionally, owing to the additional layers of complexity they presented, radial and layered node-link diagrams are hypothesized to be more difficult to learn compared to the straight-line layout. However, as these additional layers should allow for certain tasks to be answered more easily, we anticipate them to be easier to use.

4.3.2. Quantitative evaluation

We collect each participant's task answer, task completion times, and user experience questionnaire results. Specifically, for each task, participants' performance was measured using the number of errors made based on their given answers. For example, for T_{alters} or T_{edges}, the error is defined as the difference between the true number of such nodes/edges and the participant's answer. Alternatively, for task $T_{association}$, i.e. identifying the most strongly associated 1-alter with the ego, the error is defined as the difference in edge weights between the participant's answer and the ego, and the ego and the actually closest node. To statistically analyze the impact of ego network representation on these three quantities, we employ Wobbrock et al.'s [66] non-parametric aligned rank-transformed ANOVA, as standard assumptions of normality could neither be made nor validated when probed with Shapiro–Wilk tests. The overall statistical significance of ego network representations on these quantities is first probed with an omnibus *F*-test, which, if significant, is followed by a series of pairwise *t*-tests between individual representations using the computed estimated marginal means [67], both at an *a priori*, Bonferroni-adjusted family-wise type-I error rate of $\alpha = 0.05$ [68].

4.3.3. Qualitative evaluation

We collect qualitative feedback from participants in the form of open questions after every task and summarily at the end of the session. Given the 120 participants, six task-related questions, and one summary question, 840 qualitative comments are to be collected. These comments are then broken up further into individual utterances. We analyze these utterances in inductive and deductive coding sessions [69], with three independent coders. In the first inductive coding session, each

coder assigns a single concept to each utterance. These are ultimately unified into a single set of unique codes. In a second, deductive coding session, every coder assigns one of the thus agreed-upon codes to each utterance. Coders also assign a positive or negative qualifier to every utterance. In a final discussion, each utterance's coding is discussed until 100% consensus is reached.

5. Results

In this section, we enumerate the quantitative and qualitative results collected on user performance and experience.

5.1. Participants

In order to evaluate the impact of ego network representation on both graph analysis task completion time and accuracy, as well as user experience, a large-scale, interactive online user study was conducted on the user recruitment platform Prolific [70]. Each participant was randomly assigned to one of the four representations, provided said representation had not already received its total number of participants, guaranteeing a balanced number of participants per representation. In total 120 participants, i.e. 30 per representation, were recruited, of which 60 identified as "female", 59 as "male", and 1 as "other". Each participant was paid 10£ per hour, slightly above Prolific's recommended hourly rate of 9£ per hour. In terms of age, 38 participants were between the ages of 21–25, 42 between 26–30, 21 between 31–35, 7 between 36–40, 3 between 41–45, 2 between 46–50, and 6 older than 50. Given the academic bent of both the conducted literature survey as well as the selected tasks, we opted to (insofar possible) target a non-layperson participant group, i.e. people who had already completed or were in the process of completing some form of higher education. As such, 8 participants were in the process of completing their bachelor's degree, 75 participants had completed their bachelor's degree, and 37 had completed their master's degree. Finally, in terms of self-described previous experience with graph visualizations and analysis, 34 participants reported "no experience", 36 "little experience", 34 "some experience", and 16 "good experience". When probed statistically using Wobbrock et al.'s previously discussed *ART-ANOVA* [66], no significant association between the expertise of users and their performance was found.

5.2. User performance

Accuracy. For each task, participants' performance was measured using the number of errors made, depending on the user's answers and the particular task. Six answers in total had to be manually removed, such as lists of nodes when the question only asked for one, or long textual descriptions that did not actually provide an answer. The calculated error rates are depicted in Fig. 6. Here, of the six tasks analyzed, we found the impact of ego network representation on performance to be statistically significant for four of them, namely T_{common}, $T_{neighbors}$, T_{edges}, and T_{alters}. For these statistically significant tasks, representations were then compared pairwise. Here, for T_{common}, we observe adjacency matrices (median(■) = 0.35) to produce statistically significantly worse performance than both straight-line (✕ = 0.0) and radial node-link (● = 0.06) representations. For $T_{neighbors}$, the adjacency matrix (■ = 0.06) was also statistically significantly less accurate compared to all other representations (✕ = ● = ▲ = 0.0). For T_{edges}, participants using adjacency matrices (✕ = 0.7) were statistically significantly less accurate than those using straight-line (✕ = 0.33) and layered node-link diagrams (▲ = 0.2). Finally, despite notable differences observable visually in Fig. 6 for T_{alters}, the only statistically significant pairwise differences observed were between adjacency matrices (■ = 0.1) and layered node-link diagrams (▲ = 0.0). Please refer to the supplement's Table 1 for a complete account of our results.
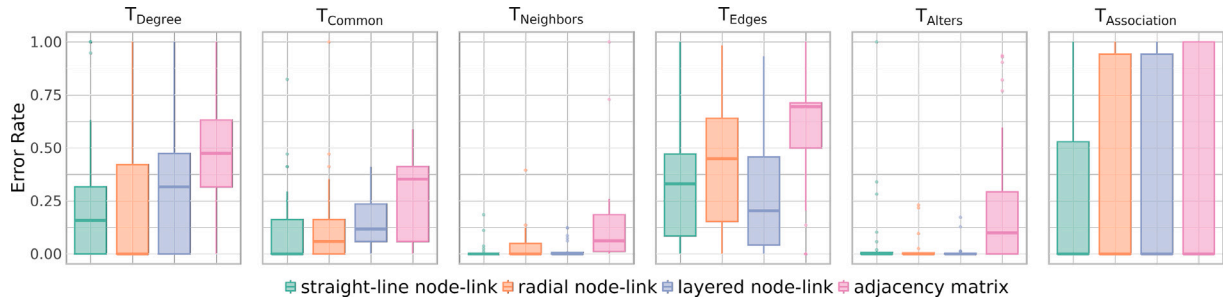
Fig. 6. Participants' task error rates are visualized per task and represented as a box-and-whisker plot. The center of each boxplot corresponds to the median, and its lower and upper hinges to the first and third quartiles, i.e. the 25th and 75th percentiles, respectively. Finally, the "whiskers" correspond to $1.5 \times IQR$ from the hinge, where IQR denotes the inter-quartile range, i.e. the distance between the first and third quartiles.
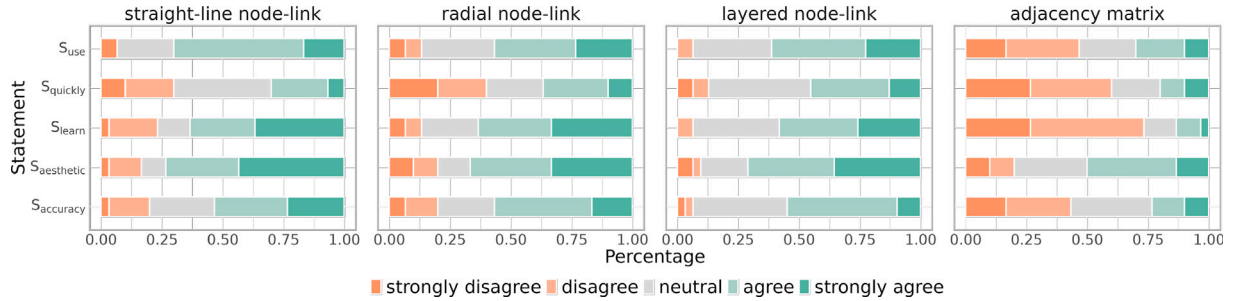

Fig. 7. 5-Point Likert Scale results visualized as normalized stacked bar chart. Each statement corresponds to a particular row, across the individual subplots corresponding to the four archetypal ego network representations investigated. Likert scale answers are color-coded, i.e. strongly disagree, disagree, neutral, agree, and strongly agree.

Time taken. The time needed to complete the survey varied between participants. At the extremes, the fastest participant completed the entire tutorial and survey in 9 min, whereas the slowest took 90 min. Most participants, i.e. 50%, however, took between 20 and 35 min, with a median time taken of 25 min. More importantly, we also track the time taken for each participant to complete each task, excluding instructions and training. Here, differences in time taken per task are fairly small and we refer to the supplementary material's Figure 2 for details. The only task for which the ego network representation proved to have a statistically significant impact was T_{Alters}. In the subsequently performed pairwise comparisons, only matrices were statistically significantly slower than the layered node-link diagram.

5.3. User experience

As discussed previously, at the end of the study, each participant was required to answer five statements on a five-point Likert scale. The results can be seen in Fig. 7. Here, the effect of ego network representation was only found to be statistically significant for two of these five statements, namely S_{learn}, i.e. the ease of learning a particular ego network representation, and S_{use}, i.e. the ease of using an ego network representation.

For these two statements, the marginal means of each representation are pairwise compared. For S_{learn} participants rated matrices statistically significantly lower than all three node-link diagrammatic representations. For S_{use}, adjacency matrices were rated statistically significantly lower than both layered and straight-line node-link diagrams. Please note that, as $S_{pleasing}$, $S_{accurately}$, and S_{learn} were not found to be statistically or substantively significant, they will not be further discussed in the results.

5.4. User feedback

We collected user feedback per task and summarily after all tasks were completed, resulting in a total of 840 comments, which were

ultimately broken down into a total of 1031 utterances. In the first inductive coding round, the three coders independently identified 30, 75, and 73 separate concepts. These emergent concepts were subsequently unified into 17 unique codes, and categorized into five broader classes, namely (i) layout, (ii) task load, (iii) interactivity, (iv) comprehension, and (v) graph tasks. In the subsequent deductive coding step, the coders individually assigned one of the 17 agreed-upon codes to each of the utterances, ignoring utterances that held no value or were incomprehensible. Additionally, each utterance was also labeled as either positive or negative, denoted here as (*pos*, *−neg*). Ultimately, in a final meeting, the three coders discussed their choices for each utterance until a consensus for every one of the 837 remaining utterances was reached, the results of which are presented in Fig. 8.

Layout, the largest class of codes with 299 total utterances, contains sentiments related to the four visual embeddings, i.e. the use of "*Color*" for nodes and edges, the chosen "*Edge Weight Encoding*", node "*Labels*" and "*Layers*", as well as the representations' "*Edge Placement*" and "*Node Placement*". "*Colors*" (105, −5) and "*Edge Weight Encoding*" (56, −7) used in the graphs were regarded generally positive. "*Node Placement*" (7, −19), "*Edge Placement*" (8, −47), and "*Labels*" (3, −5) were often mentioned in connection with visual clutter. Representations using "*Layers*" to separate alter levels were often mentioned in a positive manner (27, −1). Interestingly, the "*Visual Appeal*" of individual representations was sparsely mentioned (8, −1).

Task Load was broken down into three codes derived from the well-known NASA TLX questionnaire [71]. Notably, however, "*Physical Demand*", "*Effort*", and "*Performance*" were omitted as utterances related to them could not be identified. In total, 215 utterances related to task load were identified. The largest portion of these utterances describe the "*Cognitive Demand*" that the participants perceived (104, −78) when performing the six low-level graph tasks. The remaining comments in this category describe the participants' subjective "*Time Demand*" (3, −16), or general "*Frustration*" with the task and representation (3, −11).

The **Comprehension** class consists of codes relating to the participants' general experience in the study. In total, 126 utterances are
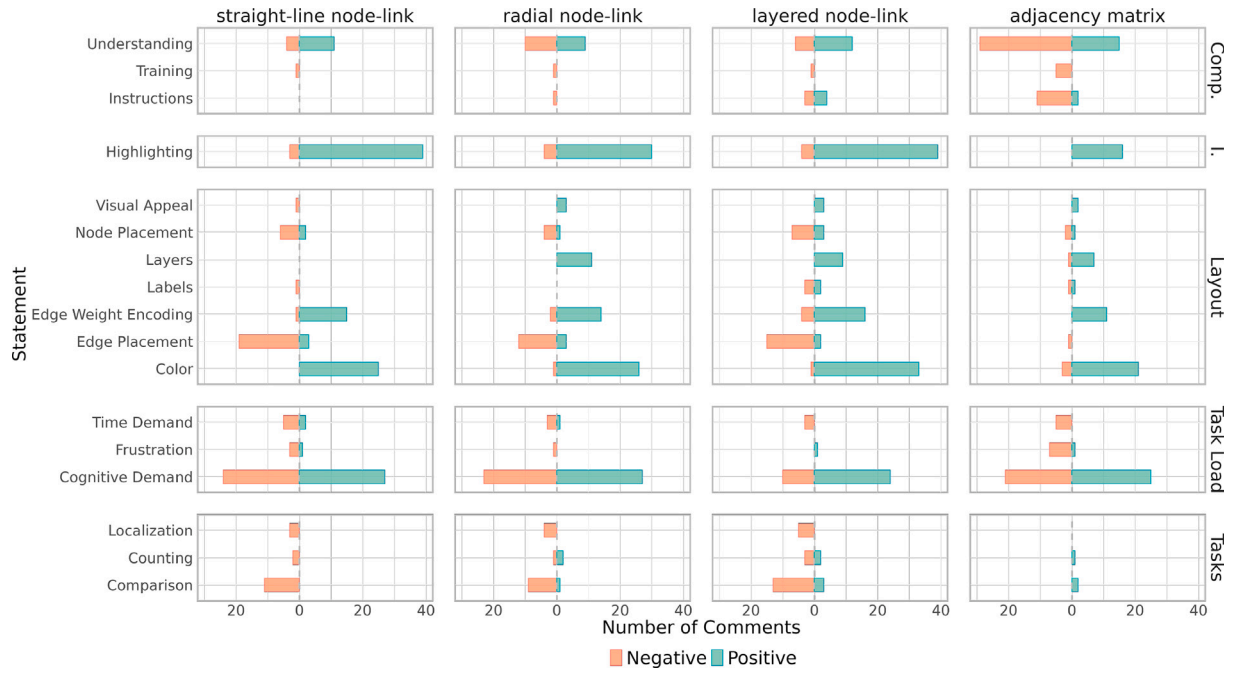
Fig. 8. The number of negative and positive statements made by participants relating to (i) comprehension, labeled as "Comp.", (ii) nodes of interaction, labeled as "I"., (iii) the layout, (iv) participant's perceived task load, and (v) particular graph tasks, labeled as "Tasks", organized by each of the four ego network visualizations. Comment counts by statement type are color-coded based on whether they are negative or positive.

mapped to this class. Several utterances highlighted participants' (lack of) "*Understanding*" as a result of the representation (47, −50). Several participants lamented problems with the clarity of the "*Instructions*" (6, −15), as well as the insufficiency of the "*Training*" (−8) that was provided.

The **Graph Tasks** class comprised codes referring to challenges with specific low-level graph tasks, totaling a sum of 62 utterances. Several participants had problems with the "*Comparison*" of two nodes' adjacencies (6, −33), the "*Localization*" of specific nodes in the graph drawing (−12), or the "*Counting*" of nodes and edges (5, −6).

Finally, the **Interaction** class was made up of solely a single code, namely the interactive "*Highlighting*" of nodes and, in the case of the adjacency matrix, edges. In total, 135 utterances are mapped to this particular class. In general, comments were overwhelmingly positive (124), with only a few comments left complaining about the implementation of interaction (−11).

6. Discussion

In the following section, we discuss the quantitative and qualitative differences between straight-line node-link diagrams (✕), radial node-link diagrams (●), layered node-link diagrams (▲), and adjacency matrices (■) within the broader context of related literature.

6.1. Matrix mayhem

As previously discussed, (conflicting) results of previous quantitative comparisons of participants' performance between adjacency matrices and node-link diagrams indicate that (i) observed differences are highly task-dependent, and (ii) for many tasks no clear "winner" can be determined. Here, we discuss this study's observed differences in participant performance across the six tasks investigated. Additionally, we also discuss differences in understanding that participants exhibited between adjacency matrices and node-link diagrams.

A surprisingly systematic subpar showing. The tabular nature of adjacency matrices, i.e. its linear arrangement of nodes and non-obscured display of edges, has been speculated (and partially shown) to make certain tasks easier, namely the localizing of nodes, based on identifier and degree, as well as the general estimation of a node's degree and the counting of edges, respectively [9,13]. It is subsequently surprising that our results do not agree with the results of related work on adjacency matrices. Specifically, for the overview (T_{edges} and T_{alters}) and adjacency (T_{degree} and $T_{neighbors}$) tasks, we hypothesized that the tabular nature and layered grouping of alter-levels in the adjacency matrix representations would be advantageous. Participants using layered adjacency matrices either performed equivalently to those using node-link diagrams (T_{alters}) or systematically statistically significantly worse (T_{edges}, T_{degree}, and $T_{neighbors}$). These quantitative results indicate that we are unable to produce the evidence necessary to confirm (parts of) our performance-oriented hypotheses stated in H_1.

Looking at the feedback provided by participants (see Fig. 8), we note some interesting differences that provide insights into these quantitative results. For example, as hypothesized by Ghoniem et al. [13] and categorized here under "*Understanding*", we noted several (11) participants explicitly voice their confusion when (double) counting (intra/inter-layer) edges in the layered adjacency matrix representation during task T_{edges}, possibly explaining the observed poor participant performance for this specific task. Additionally, despite the differences in user performance (see Fig. 6) across tasks, it is interesting that participants were critical of all three node-link diagrammatic representations, specifically their "*Edge Placement*" (✕: −19, ●: −12, ▲: −15), and to a lesser extent "*Node Placement*" (✕: −6, ●: −4, ▲: −7). This was not the case for the adjacency matrix, i.e. (■: −1) and (■: −2), respectively (Fig. 8). This gives credence to the notion that matrices are possibly preferable for larger and denser graphs owing to the absence of edge and node occlusions [13,19,22]. Similarly, the adjacency matrix participants were also less critical of both node "*Localization*" (■: 0)

and neighbor "*Counting*" (■: 0) tasks (Fig. 8) compared to those using node-link diagrams, i.e. (✕: −3, ●: −4, ▲: −5) and (✕: −2, ●: −1, ▲: −3) respectively. This further highlights the previously theorized advantages of adjacency matrices [9,13]. Nonetheless, at least for the chosen graph data, provided training, and user group, these conceptual advantages of adjacency matrices were not beneficial enough to positively impact participant performance for the overview and adjacency tasks. A possible explanation, based on the self-reported user experience (see Fig. 7), could be the fact that adjacency matrices were statistically significantly harder to learn and use, i.e., S_{Learn} and S_{Use} respectively, compared to their node-link diagrammatic counterparts. This, coupled with the disproportionate number of comments illustrating the poor "*Understanding*" of participants using adjacency matrices (■: −29), compared to the three node-link diagrams (✕: −4, ●: −10, ▲: −6), indicates that participants were not fully equipped to utilize adjacency matrices, despite their having received equivalent training.

A friend of a friend of a friend. While the tabular nature of adjacency matrices can make certain tasks conceptually simpler, it can also make others more difficult, such as the investigated common neighbor (T_{common}) and accessibility ($T_{association}$) tasks. In our study, in line with inconclusive related literature [19,26,30], we demonstrate (i) the statistical inferiority of adjacency matrices to straight-line and radial node-link diagrams for T_{common}, and (ii) the statistical inferiority of adjacency matrices to layered node-link diagrams for $T_{association}$. This, in turn, provides partial evidence for our hypotheses stated in H_3.

Looking at the results of $T_{association}$ more closely (see Fig. 6), it is notable how many participants, especially those using adjacency matrices, failed to select a 1-alter node, thereby incurring a maximal error. On the one hand, participants found adjacency matrices statistically significantly more difficult to learn and use. On the other hand, many participants positively and correctly described the representation's "*Edge Weight Encoding*" (■: 11) and its utility, thereby clearly demonstrating an understanding of the task and representation. Perhaps these observations highlight the divide between those participants demonstrating a good and bad "*Understanding*" of the representation, i.e., (■: 15) and (■: −29), respectively. It is also possible that the "*Cognitive Demand*" (■: −21) was high for the participants using adjacency matrices, especially for the complex T_{common} task for which five adjacency matrix users explicitly voiced the high cognitive demand.

Visible confusion. Looking at user experience (see Fig. 7), participants found adjacency matrices both statistically significantly more difficult to learn (compared to all three node-link diagrammatic representations) and use (compared to layered and straight-line node-link diagrams). Looking at the provided user comments (see Fig. 8), the most striking difference between adjacency matrices and all three node-link diagrams across all tasks investigated, is the many negative comments left regarding participant "*Understanding*", i.e. (■: −29) and (✕: −4, ●: −10, ▲: −6) respectively. Here, previous comparisons of straight-line node-link diagrams and adjacency matrices have speculated that differences in performance may be, at least partially, attributable to participants' lack of *a priori* familiarity and experience with adjacency matrix representations of graphs [20,23]. Moreover, past studies have shown that extensive training [10] as well as gaining familiarity over the course of a longer study [25] positively impacted participant performance using adjacency matrices. In line with these previous studies, participants using adjacency matrices did respond more critically regarding the provided "*Training*" (■: −5) and "*Instructions*" (■: −11) compared to those using the three node-link diagrams, i.e. (✕: −1, ●: −1, ▲: −1) and (✕: 0, ●: −1, ▲: −3) respectively. This could indicate that longer and more in-depth pre-study training could have assisted participants in better understanding the presented layered adjacency matrix representation. This, in turn, could have assisted them in overcoming the hypothesized *a priori* unfamiliarity with the representation, thereby allowing them also to perform better.

6.2. The holy node-link trinity

Here, given the previously enumerated quantitative and qualitative results, we discuss the differences between the investigated node-link diagrams.

History repeating itself. In line with previous studies conducted comparing different node-link diagram representations and layouts [11,12], we observe a statistically significant effect of network representation on user performance generally, namely for T_{common}, $T_{neighbors}$, T_{edges}, and T_{alters}, but ultimately cannot detect any statistically significant differences between individual node-link diagrammatic representations. Additionally, differences in the time taken per task proved to be fairly small. Specifically, for the ego network representation only one task had a significant impact on the time taken, namely for T_{alters} and, again, no significant pairwise differences between the three node-link diagrammatic representations could be found. Beyond statistical evaluations, visual inspection of participant performance (see Fig. 6) does not reveal any meaningful trends in the differences between these three node-link representations either. Ultimately, it would appear as though we must concur with past evaluations [11,12], i.e. differences between different node-link diagrams are present though very small and are not statistically detectable. This also means we cannot find evidence for our hypotheses of H_2 and H_3. Similarly, it is interesting how, for most categories (Fig. 8), all three node-link diagrams proved fairly comparable, with some particular exceptions to be discussed below.

Structural speculation. Looking at participants' comments across the six tasks (Fig. 8), it is notable how, for both the radial and layered representations, "*Layers*" were mentioned exclusively positively, i.e. (●: 11) and (▲: 9) respectively. This indicates that the partitioning of nodes, in addition to their "*Color*"-coding (✕: 25, ●: 26, ▲: 33), by alter-level was indeed helpful to participants. Relatedly, negative comments regarding the three representations' "*Edge Placement*" were slightly more frequent for the straight-line node-link diagram (✕: −19) than either the radial (●: −12) or layered node-link diagram (▲: −15), potentially further highlighting the utility of a more structured representation. Here, we speculate that the more structured and straightforward representation of the layered node-link diagram could explain the difference in negative "*Cognitive Demand*" comments left (▲: −10), compared to node-link (✕: −24) and radial (●: −23) representations.

Experiential escapades. Looking at differences in user experience, it is interesting that the ego network representation only had a statistically significant impact on participants' self-reported ease of use, S_{use}, and ease of learning, S_{learn}. However, for those two statements, there were no statistically significant pairwise differences between node-link diagrams, which, in turn, means that we are unable to provide evidence for our hypothesis H_4. Looking at the qualitative feedback provided by participants, we note some interesting visual observations. As discussed previously, while participants were positive regarding the use of "*Color*" across node-link representations, it is notable how frequently participants commented positively about radial and layered node-link diagrams' "*Layers*". Specifically, these comments often described how helpful these layers were in distinguishing alter levels or inter- and intra-alter edges from each other. This could explain the minor visual differences between layered and the other two node-link diagrammatic representations across statements (see Fig. 7). For S_{learn} this could indicate that layers assisted participants in understanding the fundamentally layered structure of ego-networks. This, to a lesser extent, could also explain the visual differences observed in $S_{quickly}$ and $S_{accuracy}$. At least for the data presented, these hypothesized benefits did not translate to statistically significant differences in user experience.

6.3. Increasingly interactive illustrations

While not directly related to the representations investigated, many participants left positive comments on the "*Interactive Highlighting*" that each visual representation featured, i.e. 124 in total. These comments were especially numerous across the three node-link diagrammatic representations (✕: 39, ●: 30, ▲: 39) compared to the adjacency matrix (■: 16), pointing out that certain tasks would have been much more difficult or even impossible without such interaction.

We note several negative comments regarding user interaction, often regarding the specific graph tasks, i.e. "*Localization*" (total = −12), "*Counting*" (total = −6), and "*Comparison*" (total = −33). Again, such comments were more numerous for the three node-link diagrammatic representations, with no single negative utterance across these categories from users of the adjacency matrix. Participants often voiced their desire for some form of automation to make such tasks less complex or arduous. They often wished for certain features such as the ability to filter nodes and edges, a node-lookup feature, the automatic counting of a node's neighbors, or the ability to select and compare multiple nodes at once. Coupled with some of the previously discussed difficulties participants had answering certain questions and the subsequently increased "*Cognitive Demand*" required, these results further point towards the need for meaningful interaction and automated analysis tools in (seemingly especially in node-link diagrammatic) ego network visualizations designed for application and production [72–74].

6.4. It's not you, it's me

Based on previous work, particularly non-ego-network comparisons of adjacency matrices and various node-link diagrams (Section 2.1), we formulated a series of hypotheses regarding the performance and preferences of users (Section 4.3.1). However, we were unable to find evidence for all posited hypotheses. Most notably, we expected adjacency matrices to perform *better* than the three node-link diagrammatic representations for all tasks (H_1) but $T_{association}$ and T_{common} (H_3). We found our layered adjacency matrices to perform (statistically significantly) worse across all six tasks (Fig. 6). We posit that, in addition to the usual difficulties of learning and understanding adjacency matrices (Section 6.1), the complexity introduced by an ego network's alter layers exacerbated these issues. While the layered structure of the implemented layered ego network adjacency matrix *should* have made certain tasks easier, it was even more complex to understand than conventional, non-ego-network adjacency matrices. Given the limited work on centered/layered adjacency matrices, it is difficult to relate these findings to previous work. However, in Bae et al.'s [53] comparison of centered adjacency matrices, quilts, and node-link diagrams, adjacency matrices consistently required more time of participants to complete the low-level graph tasks presented. As also speculated by Bae et al. [53], this could indicate that certain tasks are indeed harder to complete with a layered adjacency matrix than other embedding types. The layered structure of ego networks appears to have been understood much better in both radial and layered node-link diagrams. This indicates that more research in this direction is necessary to fully understand how an ego network's layered structure can be most effectively communicated in adjacency matrices specifically and what lessons can be learned from node-link diagrams.

7. Limitations and future work

Layout limitations. In this study, in the interest of comparability across the three node-link diagrammatic representations and (layered) adjacency matrix, we have focused on one particular set of visual mappings implemented in *D3.js* (Section 3). Given its preattentive properties, we

selected color as the visual variable to encode alter levels, i.e. embedded node attributes [51], and (where possible) node position, as well as edge weight using line segment opacity. However, several visual channels were not utilized at all, such as node size, node shape, or edge thickness, which could have additionally been utilized to further communicate topological properties. For example, alter/ego degree could have been communicated using node size. Alternatively, alter level could have been redundantly encoded in the nodes' shapes. Here, in both the interest of simplicity and comparability to adjacency matrices, such additional mappings were not considered. As our design choices would have impacted a user's interpretation of the data and identification of patterns, we opted to keep these visual attributes constant across all representations.

Future work as well as application-driven implementations of the network representations featured here should make appropriate use of the visual channels to communicate as effectively as possible the necessary topological information. For example, redundantly encoding alter level with node color and shape could allow for greater accessibility for color-blind users. Additionally, layout approaches outside of *D3.js*' force-based layout should be considered, such as Sugiyama-based algorithms [75] for layered node-link diagrams, or frameworks akin to *Circos* [76] for radial node-link diagrams. We opted to utilize *D3.js* for all representations to ensure aesthetic and run-time comparability across all representations. Future research endeavors and application-driven implementations should consider and investigate further these alternative layout approaches.

Bringing out the worst in you. Given the hypothesized and shown poor scalability of node-link diagrams (compared to adjacency matrices) [9, 19,32,77,78] and the many negative comments left regarding node-link diagrams but not adjacency matrices, it is interesting that (i) participants using adjacency matrices performed systematically worse and that (ii) no statistical differences between node-link diagrammatic representations could be detected. We speculate that the selected network size and density may have played a role in this. Increased numbers of nodes and edges could render straight-line node-link diagrams increasingly unreadable, complicating counting, adjacency, and look-up tasks. On the other hand, the orderly arrangements of nodes in radial and layered node-link diagrams, as well as adjacency matrices, could mitigate these effects. We, therefore, recommend that future works investigates the differences in participant performance and experience, taking also into account graphs of greater sizes and complexities with different selected egos in an even larger follow-up user study.

Practice, practice, practice. Participants appear to have had less of an "*Understanding*" of the representation and tasks when using adjacent matrices (Fig. 8); an observation underscored by their systematically poor performance. It is worth asking to what extent this observation is owed to the representation itself, the training and instructions participants received, or, as hypothesized in previously conducted studies [13, 20,23,79], participants' *a priori* lack of familiarity with adjacency matrices. Given previous studies highlighting the benefits of more extensive training [10,25] as well as the participants' negative comments on using the adjacency matrices related to "*Training*" and "*Instructions*", we wonder whether our results would have painted a more favorable picture, had we employed longer and more in-depth training. We argue that, despite participants' poor performance using adjacency matrices, the previously discussed conceptual advantages of adjacency matrices over node-link diagrams still hold and are a direction worth pursuing. Future research efforts should continue to investigate this representation as a potentially powerful alternative to node-link diagrams. To make such adjacency representations more useful, it would be interesting to thoroughly investigate the measurable effect of different types of training on participant performance when using adjacency matrices.

Such results would be invaluable in guiding researchers and engineers alike in selecting their ego network representation and choosing appropriate methods of training.

Not my type. In this paper, we focus exclusively on an undirected, weighted ego network. While the majority of the findings presented here should hold for unweighted graphs as well, not all do. Most notably, the tie-breaking mechanism of the layered adjacency matrix requires weights to determine the order of the ego network's nodes within each k-alter block (Section 3.4). Future work should certainly make an effort to study the impact these visual representations could have on different types of networks, which feature directed or unweighted edges, multiple node or edge types, (hierarchical) group structures, or multivariate attributes that need to be displayed.

Merely a first look. While great care was taken in selecting a representative dataset (Section 4.1), it must naturally be acknowledged that a single dataset cannot fully capture the complexities of all possible ego networks. For example, one could have selected multiple different egos to investigate within the investigated "*Les Misérables*" dataset, resulting in different maximum values of k, different topologies, and subsequently different network visualizations. Alternatively, multiple different datasets of different node sizes $|N|$ and edge numbers $|E|$ could be investigated, each with its own set of ego, k, and visualizations. However, within the context of the labor-intensive mixed methods analysis conducted here, such larger-scale studies were beyond the capabilities of this first systematic look at different layout approaches to ego networks. Future work should entertain investigating such scenarios (perhaps purely quantitatively) to paint a fuller picture of when these different visualizations are to be put to best use.

8. Conclusion

In this paper, we compile a list of 50 ego network visualizations in order to identify the most common approaches, i.e. straight-line, radial, and layered node-link diagrams, as well as adjacency matrices. We then study these approaches' quantitative and qualitative impact across six different ego network-specific graph analysis tasks on user performance and user experience in a large-scale, crowd-sourced user study of 120 participants on a single, intermediate-sized, representative graph dataset. Our results indicate that:

1. Participants using adjacency matrices performed systematically worse than those using node-link representations, despite the former's conceptual advantages highlighted both in literature and in the participants' comments.
2. In line with previous studies, all three node-link diagrams performed very similarly to each other, despite the many positive comments highlighting the conceptual benefits of alter-layers on both user performance and learning.
3. The representation had hardly any impact on the time participants needed to complete the six tasks investigated.
4. There is a need for greater training and appropriate instructions in order to overcome participants' lack of familiarity with adjacency matrix representations in particular.

Ultimately, unless an ego network's user group is already familiar with or receives extensive training on a particular representation, we recommend the use of a layered node-link diagram, as these proved statistically comparable to other node-link diagrams and superior to adjacency matrices in terms of both user performance and experience. Notable differences in user experience and comments left by our study participants hint at the value of the layered node-link diagram's more structured and straightforward representation of the ego, its alters, and their inter and intra-alter relationships, compared to radial and straight-line node-link diagrams.

CRediT authorship contribution statement

Henry Ehlers: Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Daniel Pahr:** Writing – review & editing, Formal analysis, Conceptualization. **Velitchko Filipov:** Writing – review & editing, Visualization, Methodology. **Hsiang-Yun Wu:** Writing – review & editing, Supervision. **Renata G. Raidou:** Writing – review & editing, Supervision, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cag.2024.104123.

Data availability

Data will be made available on request.

References

[1] Wu Y, Pitipornvivat N, Zhao J, Yang S, Huang G, Qu H. egoSlider: Visual analysis of egocentric network evolution. IEEE Trans Vis Comput Graphics 2016;22(1):260–9. http://dx.doi.org/10.1109/TVCG.2015.2468151.

[2] Ezaiza H, Humayoun SR, AlTarawneh R, Ebert A. PerSoN-Vis: Visualizing personal social networks (ego networks). In: Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems. CHI EA '16, New York, NY, USA: Association for Computing Machinery; 2016, p. 1222–8. http://dx.doi.org/10.1145/2851581.2892389.

[3] Lu Q, Huang J, Ge Y, Wen D, Chen B, Yu Y. EgoVis: A visual analysis system for social networks based on egocentric research. Int J Coop Inf Syst 2020;29(01n02):1930003. http://dx.doi.org/10.1142/S0218843019300031.

[4] Pu J, Zhang J, Shao H, Zhang T, Rao Y. egoDetect: Visual detection and exploration of anomaly in social communication network. Sensors 2020;20(20):5895. http://dx.doi.org/10.3390/s20205895.

[5] Liu D, Guo F, Deng B, Qu H, Wu Y. Egocomp: A node-link-based technique for visual comparison of ego-networks. Inform Visual 2017;16(3):179–89. http://dx.doi.org/10.1177/1473871616667632.

[6] Al-Awami AK, Beyer J, Strobelt H, Kasthuri N, Lichtman JW, Pfister H, et al. NeuroLines: A subway map metaphor for visualizing nanoscale neuronal connectivity. IEEE Trans Vis Comput Graphics 2014;20(12):2369–78. http://dx.doi.org/10.1109/TVCG.2014.2346312.

[7] Sayers C. Node-centric RDF graph visualization. 2004, URL https://api.semanticscholar.org/CorpusID:17594791.

[8] Zhao J, Glueck M, Chevalier F, Wu Y, Khan A. Egocentric analysis of dynamic networks with EgoLines. In: Proceedings of the 2016 CHI conference on human factors in computing systems. New York, NY, USA: Association for Computing Machinery; 2016, p. 5003–14. http://dx.doi.org/10.1145/2858036.2858488.

[9] Ghoniem M, Fekete J-D, Castagliola P. A comparison of the readability of graphs using node-link and matrix-based representations. In: IEEE symposium on information visualization. 2004, p. 17–24. http://dx.doi.org/10.1109/INFVIS.2004.1.

[10] Nobre C, Wootton D, Harrison L, Lex A. Evaluating multivariate network visualization techniques using a validated design and crowdsourcing approach. In: Proceedings of the 2020 CHI conference on human factors in computing systems. New York, NY, USA: Association for Computing Machinery; 2020, p. 1–12. http://dx.doi.org/10.1145/3313831.3376381.

[11] Huang W, Hong S-H, Eades P. Layout effects on sociogram perception. In: Healy P, Nikolov NS, editors. Graph drawing. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2006, p. 262–73. http://dx.doi.org/10.1007/11618058_24.

[12] Meulemans W, Schulz A. A tale of two communities: Assessing homophily in node-link diagrams. In: Di Giacomo E, Lubiw A, editors. Graph drawing and network visualization. Lecture notes in computer science, Cham: Springer International Publishing; 2015, p. 489–501. http://dx.doi.org/10.1007/978-3-319-27261-0_40.

[13] 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. Inform Visual 2005;4(2):114–35. http://dx.doi.org/10.1057/palgrave.ivs.9500092.

[14] Lee B, Plaisant C, Parr CS, 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 Inform Visual. New York, NY, USA: Association for Computing Machinery; 2006, p. 1–5. http://dx.doi.org/10.1145/1168149.1168168.

[15] Burch M, Huang W, Wakefield M, Purchase HC, Weiskopf D, Hua J. The state of the art in empirical user evaluation of graph visualizations. IEEE Access 2021;9:4173–98. http://dx.doi.org/10.1109/ACCESS.2020.3047616.

[16] Henry N, Fekete J-D. MatLink: Enhanced matrix visualization for analyzing social networks. In: Baranauskas C, Palanque P, Abascal J, Barbosa SDJ, editors. Human-computer interaction – INTERACT 2007. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2007, p. 288–302. http://dx.doi.org/10.1007/978-3-540-74800-7_24.

[17] Di Giacomo E, Didimo W, Montecchiani F, Tappini A. A user study on hybrid graph visualizations. In: Purchase HC, Rutter I, editors. Graph drawing and network visualization. Cham: Springer International Publishing; 2021, p. 21–38. http://dx.doi.org/10.1007/978-3-030-92931-2_2.

[18] Giacomo ED, Didimo W, Liotta G, Montecchiani F, Tappini A. Comparative study and evaluation of hybrid visualizations of graphs. IEEE Trans Vis Comput Graphics 2024;30(7):3503–15. http://dx.doi.org/10.1109/TVCG.2022.3233389, Conference Name: IEEE Transactions on Visualization and Computer Graphics.

[19] 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. New York, NY, USA: Association for Computing Machinery; 2013, p. 483–92. http://dx.doi.org/10.1145/2470654.2470724.

[20] Berardi CW, Solovey ET, Cummings ML. Investigating the efficacy of network visualizations for intelligence tasks. In: 2013 IEEE international conference on intelligence and security informatics. 2013, p. 278–83. http://dx.doi.org/10.1109/ISI.2013.6578843.

[21] Okoe M, Jianu R, Kobourov S. Revisited experimental comparison of node-link and matrix representations. In: Frati F, Ma K-L, editors. Graph drawing and network visualization. Lecture notes in computer science, Cham: Springer International Publishing; 2018, p. 287–302. http://dx.doi.org/10.1007/978-3-319-73915-1_23.

[22] Okoe M, Jianu R, Kobourov S. Node-link or adjacency matrices: Old question, new insights. IEEE Trans Vis Comput Graphics 2019;25(10):2940–52. http://dx.doi.org/10.1109/TVCG.2018.2865940.

[23] Chang C, Bach B, Dwyer T, Marriott K. Evaluating perceptually complementary views for network exploration tasks. In: Proceedings of the 2017 CHI conference on human factors in computing systems. New York, NY, USA: Association for Computing Machinery; 2017, p. 1397–407. http://dx.doi.org/10.1145/3025453.3026024.

[24] Didimo W, Kornaropoulos EM, Montecchiani F, Tollis IG. A visualization framework and user studies for overloaded orthogonal drawings. Comput Graph Forum 2018;37(1):288–300. http://dx.doi.org/10.1111/cgf.13266.

[25] Ren D, Marusich LR, O'Donovan J, Bakdash JZ, Schaffer JA, Cassenti DN, et al. Understanding node-link and matrix visualizations of networks: A large-scale online experiment. Netw Sci 2019;7(2):242–64. http://dx.doi.org/10.1017/nws.2019.6.

[26] Keller R, Eckert CM, Clarkson PJ. Matrices or node-link diagrams: Which visual representation is better for visualising connectivity models? Inform Visual 2006;5(1):62–76. http://dx.doi.org/10.1057/palgrave.ivs.9500116.

[27] Okoe M, Jianu R. Ecological validity in quantitative user studies – a case study in graph evaluation. In: Proc. IEEE VIS. 2015.

[28] Hlawatsch M, Burch M, Weiskopf D. Visual adjacency lists for dynamic graphs. IEEE Trans Vis Comput Graphics 2014;20(11):1590–603. http://dx.doi.org/10.1109/TVCG.2014.2322594.

[29] Okoe M, Jianu R. GraphUnit: Evaluating interactive graph visualizations using crowdsourcing. Comput Graph Forum 2015;34(3):451–60. http://dx.doi.org/10.1111/cgf.12657.

[30] McIntire JP, Osesina OI, Bartley C, Tudoreanu ME, Havig PR, Geiselman EE. Visualizing weighted networks: a performance comparison of adjacency matrices versus node-link diagrams. In: Ground/air multisensor interoperability, integration, and networking for persistent ISR III. Baltimore, Maryland; 2012, p. 83891G–83891G–7. http://dx.doi.org/10.1117/12.920012.

[31] Yoghourdjian V, Dwyer T, Klein K, Marriott K, Wybrow M. Graph thumbnails: Identifying and comparing multiple graphs at a glance. IEEE Trans Vis Comput Graphics 2018;24(12):3081–95. http://dx.doi.org/10.1109/TVCG.2018.2790961.

[32] Abdelaal M, Schiele ND, Angerbauer K, Kurzhals K, Sedlmair M, Weiskopf D. Comparative evaluation of bipartite, node-link, and matrix-based network representations. IEEE Trans Vis Comput Graphics 2023;29(1):896–906. http://dx.doi.org/10.1109/TVCG.2022.3209427.

[33] Blythe J, McGrath C, Krackhardt D. The effect of graph layout on inference from social network data. In: Brandenburg FJ, editor. Graph drawing. Lecture notes in computer science, Berlin, Heidelberg: Springer; 1996, p. 40–51. http://dx.doi.org/10.1007/BFb0021789.

[34] Purchase HC. Performance of layout algorithms: Comprehension, not computation. J Vis Lang Comput 1998;9(6):647–57. http://dx.doi.org/10.1006/jvlc.1998.0093.

[35] Lee B, Parr C, Plaisant C, Bederson B, Veksler V, Gray W, et al. TreePlus: Interactive exploration of networks with enhanced tree layouts. IEEE Trans Vis Comput Graphics 2006;12(6):1414–26. http://dx.doi.org/10.1109/TVCG.2006.106.

[36] Law P-M, Wu Y, Basole RC. Segue: Overviewing evolution patterns of egocentric networks by interactive construction of spatial layouts. In: 2018 IEEE conference on visual analytics science and technology. 2018, p. 72–83. http://dx.doi.org/10.1109/VAST.2018.8802415.

[37] Fu K, Mao T, Wang Y, Lin D, Zhang Y, Sun X. DyEgoVis: Visual exploration of dynamic ego-network evolution. Appl Sci 2021;11(5):2399. http://dx.doi.org/10.3390/app11052399.

[38] Troidl J, Cali C, Gröller E, Pfister H, Hadwiger M, Beyer J. Barrio: Customizable spatial neighborhood analysis and comparison for nanoscale brain structures. Comput Graph Forum 2022;41(3):183–94. http://dx.doi.org/10.1111/cgf.14532.

[39] Li Q, Shen Q, Ming Y, Xu P, Wang Y, Ma X, et al. A visual analytics approach for understanding egocentric intimacy network evolution and impact propagation in MMORPGs. In: 2017 IEEE Pacific visualization symposium (Pacificvis). 2017, p. 31–40. http://dx.doi.org/10.1109/PACIFICVIS.2017.8031576.

[40] Brich N, Harbig T, Paz MW, Nieselt K, Krone M. ProtEGOnist: Visual analysis of interactions in small world networks using ego-graphs. Comput Graph Forum 2024. http://dx.doi.org/10.1111/cgf.15078.

[41] Shi L, Wang C, Wen Z, Qu H, Lin C, Liao Q. 1.5D egocentric dynamic network visualization. IEEE Trans Vis Comput Graphics 2015;21(5):624–37. http://dx.doi.org/10.1109/TVCG.2014.2383380.

[42] Liu Q, Hu Y, Shi L, Mu X, Zhang Y, Tang J. EgoNetCloud: Event-based egocentric dynamic network visualization. In: 2015 IEEE conference on visual analytics science and technology. 2015, p. 65–72. http://dx.doi.org/10.1109/VAST.2015.7347632.

[43] He Q, Zhu M, Lu B, Liu H, Shen Q. MENA: Visual analysis of multivariate egocentric network evolution. In: 2016 international conference on virtual reality and visualization. 2016, p. 488–96. http://dx.doi.org/10.1109/ICVRV.2016.88.

[44] Schikora C, Isemann D. InfluViz — A visualization tool for exploring and analyzing creative influence between artists and their works. In: 2017 21st international conference information visualisation. 2017, p. 336–43. http://dx.doi.org/10.1109/iV.2017.66.

[45] Sorger J, Arleo A, Kán P, Knecht W, Waldner M. Egocentric network exploration for immersive analytics. Comput Graph Forum 2021;40(7):241–52. http://dx.doi.org/10.1111/cgf.14417.

[46] Di Battista G, Eades P, Tamassia R, Tollis IG. Graph Drawing: algorithms for the visualization of graphs. 1st ed.. Prentice Hall; 1999, http://dx.doi.org/10.5555/551884.

[47] Fisher D. Using egocentric networks to understand communication. IEEE Internet Comput 2005;9(5):20–8. http://dx.doi.org/10.1109/MIC.2005.114.

[48] Farrugia M, Hurley N, Quigley A. Exploring temporal ego networks using small multiples and tree-ring layouts. 2011.

[49] Cunningham P, Harrigan M, Wu G, O'callaghan D. Characterizing ego-networks using motifs. Netw Sci 2013;1(2):170–90. http://dx.doi.org/10.1017/nws.2013.12.

[50] Nobre C, Meyer M, Streit M, Lex A. The state of the art in visualizing multivariate networks. Comput Graph Forum 2019;38(3):807–32. http://dx.doi.org/10.1111/cgf.13728.

[51] Ehlers H, Marin D, Wu H-Y, Raidou R. Visualizing group structure in compound graphs: The current state, lessons learned, and outstanding opportunities. 2024, p. 697–708.

[52] Shen Z, Ma K-L. Path visualization for adjacency matrices. In: Proceedings of the 9th joint eurographics / IEEE VGTC conference on visualization. Goslar, DEU: Eurographics Association; 2007, p. 83–90.

[53] Bae J, Watson B. Developing and evaluating quilts for the depiction of large layered graphs. IEEE Trans Vis Comput Graphics 2011;17(12):2268–75. http://dx.doi.org/10.1109/TVCG.2011.187.

[54] Bae J, Helldin T, Riveiro M. Understanding indirect causal relationships in node-link graphs. Comput Graph Forum 2017;36(3):411–21. http://dx.doi.org/10.1111/cgf.13198.

[55] Kobourov SG. Spring embedders and force directed graph drawing algorithms. 2012, http://dx.doi.org/10.48550/arXiv.1201.3011, arXiv.

[56] Bostock M, Ogievetsky V, Heer J. D³ data-driven documents. IEEE Trans Vis Comput Graphics 2011;17(12):2301–9. http://dx.doi.org/10.1109/TVCG.2011.185.

[57] Harrower M, Brewer CA. ColorBrewer.org: An online tool for selecting colour schemes for maps. Cartogr J 2003;40(1):27–37. http://dx.doi.org/10.1179/000870403235002042.

[58] Knuth DE. The Stanford GraphBase: a platform for combinatorial computing, vol. 1, New York: AcM Press; 1993.

[59] Broido AD, Clauset A. Scale-free networks are rare. Nature Commun 2019;10(1):1017. http://dx.doi.org/10.1038/s41467-019-08746-5.

[60] Yoghourdjian V, Archambault D, Diehl S, Dwyer T, Klein K, Purchase HC, et al. Exploring the limits of complexity: A survey of empirical studies on graph visualisation. Vis Inform 2018;2(4):264–82. http://dx.doi.org/10.1016/j.visinf.2018.12.006.

[61] Zhao M, Su Y, Zhao J, Chen S, Qu H. Mobile situated analytics of ego-centric network data. In: SIGGRAPH Asia 2017 symposium on visualization. New York, NY, USA: Association for Computing Machinery; 2017, p. 1–8. http://dx.doi.org/10.1145/3139295.3139309.

[62] Lu B, Zhu M, He Q, Li M, Jia R. TMNVis: Visual analysis of evolution in temporal multivariate network at multiple granularities. J Vis Lang Comput 2017;43:30–41. http://dx.doi.org/10.1016/j.jvlc.2017.03.003.

[63] Lam H, Bertini E, Isenberg P, Plaisant C, Carpendale S. Seven guiding scenarios for information visualization evaluation. Tech. rep. hal- 00723057, 2011.

[64] Joos L, Jaeger-Honz S, Schreiber F, Keim DA, Klein K. Visual comparison of networks in VR. IEEE Trans Vis Comput Graphics 2022;28(11):3651–61. http://dx.doi.org/10.1109/TVCG.2022.3203001.

[65] Purchase HC, Cohen RF, James M. Validating graph drawing aesthetics. In: Brandenburg FJ, editor. Graph drawing. Lecture notes in computer science, Berlin, Heidelberg: Springer; 1996, p. 435–46. http://dx.doi.org/10.1007/BFb0021827.

[66] Wobbrock JO, Findlater L, Gergle D, Higgins JJ. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In: Proceedings of the SIGCHI conference on human factors in computing systems. New York, NY, USA: Association for Computing Machinery; 2011, p. 143–6. http://dx.doi.org/10.1145/1978942.1978963.

[67] Elkin LA, Kay M, Higgins JJ, Wobbrock JO. An aligned rank transform procedure for multifactor contrast tests. In: The 34th annual ACM symposium on user interface software and technology. New York, NY, USA: Association for Computing Machinery; 2021, p. 754–68. http://dx.doi.org/10.1145/3472749.3474784.

[68] Bonferroni C. Teoria statistica delle classi e calcolo delle probabilita. Pubbl del R Ist Superiore di Sci Economiche e Commericiali di Firenze 1936;8:3–62, URL https://cir.nii.ac.jp/crid/1570009749360424576.

[69] Chi MT. Quantifying qualitative analyses of verbal data: A practical guide. J Learn Sci 1997;6(3):271–315. http://dx.doi.org/10.1207/s15327809jls0603_1.

[70] Prolific. London, UK: Prolific; 2014, URL https://www.prolific.com.

[71] Hart SG, Staveland LE. Development of NASA-TLX (task load index): Results of empirical and theoretical research. In: Hancock PA, Meshkati N, editors. Advances in psychology. Human mental workload, vol. 52, North-Holland; 1988, p. 139–83. http://dx.doi.org/10.1016/S0166-4115(08)62386-9.

[72] Wybrow M, Elmqvist N, Fekete J-D, von Landesberger T, van Wijk JJ, Zimmer B. Interaction in the visualization of multivariate networks. In: Kerren A, Purchase HC, Ward MO, editors. Multivariate network visualization: dagstuhl seminar #13201, dagstuhl castle, Germany, May 12-17, 2013, revised discussions. Cham: Springer International Publishing; 2014, p. 97–125. http://dx.doi.org/10.1007/978-3-319-06793-3_6.

[73] Yi JS, Kang Ya, Stasko J, Jacko J. Toward a deeper understanding of the role of interaction in information visualization. IEEE Trans Vis Comput Graphics 2007;13(6):1224–31. http://dx.doi.org/10.1109/TVCG.2007.70515.

[74] Figueiras A. Towards the understanding of interaction in information visualization. In: 2015 19th international conference on information visualisation. 2015, p. 140–7. http://dx.doi.org/10.1109/iV.2015.34.

[75] Sugiyama K, Tagawa S, Toda M. Methods for visual understanding of hierarchical system structures. IEEE Trans Syst Man Cybern 1981;11(2):109–25. http://dx.doi.org/10.1109/TSMC.1981.4308636, Conference Name: IEEE Transactions on Systems, Man, and Cybernetics.

[76] Krzywinski M, Schein J, Birol I, Connors J, Gascoyne R, Horsman D, et al. Circos: An information aesthetic for comparative genomics. Genome Res 2009;19(9):1639–45. http://dx.doi.org/10.1101/gr.092759.109.

[77] Kobourov SG, Pupyrev S, Saket B. Are crossings important for drawing large graphs? In: Duncan C, Symvonis A, editors. Graph drawing. Lecture notes in computer science, Berlin, Heidelberg: Springer; 2014, p. 234–45. http://dx.doi.org/10.1007/978-3-662-45803-7_20.

[78] Ehlers H, Villedieu A, Raidou RG, Wu H-Y. Improving readability of static, straight-line graph drawings: A first look at edge crossing resolution through iterative vertex splitting. Comput Graph 2023;116:448–63. http://dx.doi.org/10.1016/j.cag.2023.09.010.

[79] Abuthawabeh A, Beck F, Zeckzer D, Diehl S. Finding structures in multi-type code couplings with node-link and matrix visualizations. In: 2013 first IEEE working conference on software visualization. 2013, p. 1–10. http://dx.doi.org/10.1109/VISSOFT.2013.6650530.