egoComp: A node-link-based technique for visual comparison of ego-networks

Dongyu Liu¹,², Fangzhou Guo¹, Bowen Deng², Huamin Qu² and Yingcai Wu¹

Abstract
Analysis of ego-networks is a critical research problem when analyzing large-scale social networks, as an ego-network represents the social circle a person actually contacts with. One of the core tasks in ego-network analysis is visual comparison, which includes edge comparison and node comparison. Although various works have been done to support comparing normal networks and ego-networks, intuitive node comparison of two ego-networks is still challenging. In this article, we propose egoComp, an intuitive and expressive visualization technique, to analyze the node difference between two ego-networks. To preserve the latent structure of ego-network and lay emphasis on intuitiveness, our design is node-link-based (radial tree layout) and uses a side-by-side method to compare ego-networks. We design a novel storyflow-like graph layout to reveal the relationship of two ego-networks at the individual node level. Furthermore, three different layout algorithms, including origin, greedy, and assignment algorithms, are proposed to meet different user requirements. We demonstrate the effectiveness of our system through case studies and a user study and then discuss the limitations thoroughly as well as the possible solutions and potential future work.

Keywords
Ego-network, node-link diagram, node comparison, comparative analysis

Introduction
Ego-network analysis is critically important for large-scale social network analysis. Ego-networks are a kind of special social networks. Unlike common social networks, an ego-network has a particular central node designated as an ego. Other nodes in the network, direct or indirect neighbors of the ego, are called alters. An ego-network shows the relations among the ego and its neighborhood. Important insights can be obtained by comparing ego-networks of different entities when exploring and analyzing social networks.¹,² In a social network, the closeness of two people can be represented by the similarity of their ego-networks. For example, in an academic collaboration network, different collaboration patterns can be revealed by comparing two authors’ ego-networks.²,³

Although people have gradually realized the importance of ego-centric analysis, it is still a very challenging task to visually tell the difference among multiple ego-networks in a huge and dense network (i.e. large graph).⁴ This work attempts to address the challenge by introducing egoComp, a technique which focuses on the analysis of node difference of two ego-networks. The technique can be easily integrated into large-scale network analysis to facilitate visual comparison and exploration of ego-networks.

¹The State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China
²Hong Kong University of Science and Technology, Kowloon, Hong Kong

Corresponding author:
Yingcai Wu, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang, 310058, P.R. China.
Email: ycwu@cad.zju.edu.cn
Three types of visual comparisons are concluded in previous work, including juxtaposition (i.e., side by side), superposition (i.e., overlay), and explicit visual encodings. Combining different visualization forms, such as node-link diagram and adjacency matrix, with different comparison techniques, researchers have developed various graph comparison methods. However, these methods cannot be easily adapted to the ego-network data because ego-networks latently contain a tree structure. Moreover, these techniques are not tailored for comparison. For example, regarding the structure of an ego-network, the ego itself typically has a number of directly connected nodes (aka hop1 alters); these hop1 alters also have numerous directly connected nodes that are not themselves (aka hop2 alters). It is a common task to understand this kind of structure and make comparison between two different ego-networks when analyzing a social network. If we use the traditional node diagram or matrix to represent an ego-network, it would be neither easy for users to distinguish the hierarchical structure of the ego-network nor for comparison.

Thus, we design and develop an intuitive and interactive ego-network comparison system. Particularly, for the purpose of intuitively comparing two ego-networks, we propose transformed radial tree diagrams to represent ego-networks and adopt side-by-side method to facilitate the comparison. As links are one of the most common methods to highlight the relationships among multiple objects, especially in graph, we use links to represent the node relationships in two ego-networks. However, when the graph is getting larger and larger, this method suffers from a serious visual clutter problem. Hence, we propose storyflow-like links to link the nodes that appear in both ego-networks to alleviate the visual clutter problem. Furthermore, the node layout is calculated by tailored optimization algorithms to minimize the number of link crossings.

The major contributions of this work are as follows:

- A novel ego-network comparison technique, which is on the basis of radial tree diagrams and a side-by-side comparison method. Storyflow-like links are employed to intuitively show the similarity of two ego-networks.
- Two layout optimization algorithms, a greedy algorithm and an assignment algorithm, to reduce the visual clutter.
- A comprehensive evaluation, including a use case and a user study, to demonstrate the effectiveness of our technique.

This article is structured as follows: section “Related works” presents related work. Section “Task analysis” focuses on task analysis. Section “Ego comparison” lays out details of our approach, followed by sections “Use case” and “User study,” in which we demonstrate the effectiveness of our system through a use case and user study. Section “Discussion” discusses the performance and limitations of our approach. Conclusion and future work are presented in section “Conclusion.”

Related works

Ego-network is an important research topic in social network analysis. Previous studies use node-link diagrams to visualize ego-networks. Depending on the data and tasks, the layout of a node-link diagram can be modified such as in Shi et al. A timeline can be added to visualize dynamic networks, and in Namata et al., nodes within a specific distance to the ego are arranged on a circle with the corresponding radius. In our work, we use the similar layout as in Namata et al., but we apply a special layout algorithm to arrange the nodes on semicircles with the least link crossings.

Bremer and Munzner conclude that visual comparison is one of the abstract visualization tasks. The techniques are classified into three categories: juxtaposition, superposition, and explicit representation of the relationships. Juxtaposition designs present objects side by side and users have to connect objects in their mental memory. Superposition designs overlay objects on top of others. Explicit encodings calculate the relationships among objects and visualize these relationships.

With these three basic methods, a number of visualization approaches have been developed to deal with different comparison tasks. Some systems support single type of comparison techniques, such as juxtaposition of node-link diagrams, superposition of node-link diagrams and adjacency matrix, and explicit encodings of the relationships among objects. Others support multiple types of techniques, such as Collins and Carpendale and Hascoet and Dragicevic. Merging networks together requires layout adjustment and more encodings to support visual comparison, which leads to less intuition. VisLink links the same objects together in two different visualizations. However, the links are not well arranged and can result in severe visual clutter. In addition, the layouts of the visualizations to be compared are fixed. Our work is inspired by the idea of VisLink, but it can avoid the clutter problem using Storyflow-like links in our work and optimizing the network layout. Edge bundling algorithms are used to solve the clutter problem caused by the links in previous studies, but a large number of edges are merged after bundling.
Evaluating the similarity of network data is another way to compare two networks. By multidimensional scaling (MDS) or other projection algorithms, a network can be represented by a dot in a scatter plot and the distance between two dots represents the similarity of two networks. A similarity-based comparison method can characterize the difference macroscopically, but it lacks the details of the underlying objects.

Ogawa and Ma and Tanahashi and Ma present storyline to visualize the evolution of software. This technique is efficient to show the change of relationships of objects and is implemented to analyze different datasets, such as storyflow. We combine the storyflow with a node-link diagram to show the similarity of two ego-networks.

**Task analysis**

Our method aims to show the node difference in a very intuitive way. We focus on visualization of ego-networks with at most two levels (i.e. ego, hop1 alters, and hop2 alters; the terms are explained in subsection **visual design**). To derive the user requirements, we conduct in-depth interviews with two domain experts in the graph mining area (one from academia and the other one from industry). They are not the co-authors of the paper. Based on their feedback and comments as well as the literature review, we obtain the important analytical tasks regarding to node comparison between two different ego-networks:

T1: What does a large graph look like?
T2: When two egos are selected, what is the overall similarity between them?
T3: Which alters have the closest relationship with an ego?
T4: Given two different ego-networks, how many hop1 alters do they share?
T5: Given two different ego-networks, what are the exact number of different alter relationships as shown in Figure 3(b) (e.g. how many hop1 alters in the left ego-network become hop1 alters or hop2 alters (or do not exist) in right ego-network)?
T6: Given an alter in one of the ego-networks, does this alter exist in the right ego-network? If that is the case, where is the exact position of that alter?

**Ego comparison**

**System overview**

Two datasets at different levels of scalability are used in our system. One is *Les Miserables* with 77 nodes and 254 edges, describing the coappearance network of characters in novel *Les Miserables*. The other one is *Netscience* with 1589 nodes and 2742 edges, illustrating the co-authorship network of scientists working on network theory and experiments. As shown in Figure 1, our system interface mainly consists of two views, the overview and the comparison view. After users select a dataset (Figure 1(a)), a network is drawn below the overview (Figure 1(d)) using Huyifan layout algorithm to provide users with an overview of the selected dataset (T1). Additionally, users are able to zoom in and zoom out the view to further explore the graph in different levels of detail. Nodes in the overview can be chosen as an ego. When two egos are selected, users can click the buttons shown in Figure 1(c) to choose a comparison layout algorithm and then see the result in the comparison view (T2, T3, T4, T5, and T6).

**Visual design**

This section defines some important terms in this article and introduces the visual encoding in the comparison view:

LEN and REN. The ego-network on the left side is abbreviated as LEN, and the one on the right side is abbreviated as REN.

Ego. Users can pick two different nodes in the overview as the central nodes of ego-network in comparison view, the first selected one for LEN and the second one for REN. Central nodes in ego-network are designated as ego. The left ego and right ego are, respectively, colored with red and purple to visually distinguish them.

Hop1 alter. The nodes that are directly connected to the ego are hop1 alters. Hop1 alters are assigned uniformly on inner semicircle.

Hop2 alter. The nodes that are directly connected to the hop1 alters but not connected directly to the ego (i.e. not the hop1 alters) are hop2 alters. Hop2 alters are assigned uniformly on outer semicircle. Note that in this work, we focus on analyzing ego’s hop1 and hop2 alters.

Node color. LEN’s hop1 alters, which are also REN’s Hop1 or Hop2 alters (common friends), are encoded with color blue. And LEN’s hop2 alters, which is common friends of REN, are encoded with color orange (T4 and T5).

Node size. The size of nodes stands for the strength of relationship (edge weight) with ego. The larger the size, the stronger the relationship between this node and ego (T3); the relationship denotes the sum of edges weight on the path (one edge for hop1 alters and two edges for hop2 alters) whose start and end point are the alter itself and the ego (if there are more than one path, we only consider the one with maximum sum of edges weight). Notice that we do not use
links of different thicknesses to represent this kind of relationship, as it will render a serious visual clutter problem.

*Link.* It is required that the difference between two ego-networks, especially the corresponding positions of LEN’s hop1 alters in REN, should be revealed in one single view rather than through interactions; it is ineffective to compare the current visualization with the one in your mind. Therefore, we use links between LEN’s alter and REN’s alter to encode such information (T6).

*Alter angle.* The included angle (Figure 2(f) and (g)) between the perpendicular line (L1, Figure 2) and the line through the ego and this alter (L2, Figure 2), where the value of angle is from 0 to $\pi$ (from top to bottom).

*Session.* The six rectangles in the middle area are called sessions. Similar to the storyflow, links will flow into specific sessions and flow out according to their attribute. For the left top session, links starting from alters whose angle less than $1/3\pi$ will flow into it. For the right top session, links ending in alters whose angle less than $1/3\pi$ will flow through it. The middle and bottom four sessions are similar to top two.

**Alternative design**

In the following, we discuss our alternative designs to visually compare two ego-networks. As mentioned before, there are three typical ways to present two graphs: side by side, and animated views. Basically, we have two design principles to follow. The first one is to make comparison as intuitive as possible, and the second is to provide information and show difference about two graphs in a single view rather than compare the current ego-network with the one in your mind. Following these two principles, we eliminate two frequently used designs, matrix design and animated views. We only consider two node-link-diagram-based methods as shown in Figure 3.

Figure 3(a) puts two ego-networks side by side, where links between two networks mean they share the same alters. These two ego-networks are laid out in a *force-directed algorithm,* and the color of nodes has the exact same meaning as Figure 2. Although this approach preserves two-dimensional (2D) position information, it suffers from terrible visual clutter problem with the increasing number of shared alters.

Figure 3(b) merges two ego-networks into one, and glyphs are used to show eight different relationships, like (1,1) represents nodes which are hop1 alters in both ego-networks. This method is able to show difference between two ego-networks clearly in a space-efficient way, but it is not intuitive enough compared to side by side view and requires users much attention to recognize the tiny difference among these eight glyphs.
Since the two alternative designs have their own critical problems, we finally decide to use node-link-based (radial tree layout) side-by-side design for comparing two different ego-networks, in which we not only emphasize the importance of intuitiveness but also reduce the visual clutter.

**Layout algorithm**

It is obvious that the alter positions in REN will greatly influence the link crossing numbers. To reduce the number of crossings, we need to use algorithm to position the alters. We assume the layout of LEN is given by simply using breadth-first search (BFS). It means that the node visited earlier by BFS visiting order will be put in the position with smaller angle (from top to bottom). What we need to compute is the layout of REN’s alters. There are three layout algorithms for REN’s alters, where the default one is exactly the same as the one we use in LEN layout. The other two layout algorithms aim to reduce the sum of alter angle difference (equation (1)) because the lower sum means decreasing of link crossings to some extent; although the lower sum of alter angle difference is not
 Actually, we can convert this problem into an assignment problem, a fundamental combinatorial optimization problem, which can be solved by Kuhn–Munkres algorithm in $O(n^3)$ and get the optimal result.

### Use case

Academic collaboration network, a special kind of social network, has been attracting a lot of research interests. People are interested in, for example, who is the best connected scientists, in what roles are the top researchers playing in their research field, and how similar their co-authorship networks are. So many questions can be asked. Here, we describe a use case to demonstrate the effectiveness of our egoComp system using Netscience dataset, which describes the co-authorship network of scientists working on network theory and experiment (Figure 1(d)).

After selecting Netscience, we can see a big picture in Figure 1(d), showing the overview of the entire selected data including 1589 nodes and 2742 edges. A node represents a scientist and an edge linking two nodes indicates the associated two scientists have a co-authored paper. Looking at the overview, we can observe that there are a large number of clusters with various sizes. Smaller clusters of no more than five nodes are laid out in the periphery, while large clusters with multiple nodes are close to the center. Obviously, one giant cluster, the largest connected component that includes 379 scientists, appears right in the center as shown in Figure 4(c).

From Figure 4(c), we can clearly see that several scientists are playing crucial roles in linking different communities together, such as NEWMAN, HOLME, and JEONG (the nodes that are highlighted inside the blue rectangle). These nodes usually have a high degree, indicating they are probably researchers with high impacting factors. Hence, we select NEWMAN and HOLME as two egos and then use our optimal layout algorithm in the comparison view to show their relationship (Figure 4(b)). We find that HOLME and NEWMAN have a direct co-authorship, and NEWMAN’s ego-network (i.e. LEN) having more hop1 alters and hop2 alters than HOLME’s. Interestingly, all of the NEWMAN’s hop1 alters (27 in total) appear in REN, among which two alters (i.e. LEICHT and GHOSHAL) are also HOLME’s hop1 alters. From the view of REN, we can see 3 (including NEWMAN) out of 14 hop1 alters have direct co-authorships with NEWMAN. Furthermore, nearly half of HOLME’s hop2 alters are NEWMAN’s direct co-authors (hop1 alters).

Another example in Figure 4(a) shows that LEN and REN are BENJACOB and JEONG’s ego-networks, respectively. Clearly, they do not have a direct co-authorship, but share one hop1 alter named...
VICSEK. In addition, only four alters in LEN cannot be found on REN. The other 16 alters are in REN, out of which 8 alters are also JEONG hop1 alters. This suggests that JEONG has worked closely with BENJACOB.

User study

Study setup

To evaluate egoComp’s ability to facilitate comparing ego-networks, we conducted a usability study. The goal of this study is to find out whether the egoComp is helpful for users to explore and compare ego-networks. This study included 12 participants (4 females, 8 males) all of whom were either graduate or undergraduate students with computer science background. The range of participant ages is from 19 to 30 years. In total, 9 out of 12 participants had prior knowledge on graph and half are familiar with visualization. We proceeded the study in a computer lab of our school. The computer used for this user study is with a display of 1920 × 1200 resolution, I7-6700K central processing unit (CPU) @4 GHz, 16 GB memory, Nvidia GeForce GTX 960, and Windows 7 64bit installed.

At first, we briefly introduced the goals and features of egoComp and showed a detailed tutorial video of the system, where the use case mentioned in the previous section was included. Afterward, the participants were asked to freely explore the system to familiarize themselves with the features and functions of egoComp. During the process, we requested them to complete several small tasks that we prepared in advance for them and speak what they thought aloud at the same time. As these small tasks were designed to help answer the questions in the task analysis part in section “Task analysis,” doing so can help us obtain the instant feedback of the proposed visualization technique. Finally, the participants were required to answer some questions related to the system usability, visual designs and interactions, and specific functions.

Study results

Figure 5 shows the participants’ subjective ratings of the egoComp. We used a 7-point Likert scale on all questions. As shown in the figure, we organized the questions into three categories, namely, system usability, user exploration experience, and the helpfulness of

![Figure 4](image_url). Two pairs of nodes, [1316, 35] (a) and [79, 757] (b), are selected to compare in the overview (c).
each view in related tasks. From the results obtained, the participants had a very positive experience with the egoComp.

Regarding the system usability, most of the participants felt that the presented system is easy to use (average score: 6.0) and easy to learn (average score: 5.7) because the visual representations are very intuitive. All the participants deemed that the proposed visualization can fulfill most of their requirements. However, there were also few limitations in current stage. For example, some participants suggested to add a search function in both the overview and comparison view, enabling them to find the nodes of interests via the name or some other key words.

Three of them showed a strong will to further explore the more detailed information of a selected alter or ego in the application context of analyzing co-authorship network, such as the author’s background and publication information.

As for the user exploration experience, participants favored our system, commenting that the interface is very clean but useful (average score: 5.8) and the interactions are very smooth and native (average score: 5.6). In particular, 10 out of 12 persons pointed out that they were very fond of the zoom in and zoom out function provided in both the overview and comparison view, finding these functions quite useful.

For the last group of questions, the participants were required to rate whether the proposed technique was helpful for the exploratory tasks in the task analysis part. From the results, we saw that all the tasks, except for the T2, received very high average scores (> 6), indicating egoComp is indeed helpful in solving these tasks. The comments coming from the participants can also be used to further improve the current design.

**Discussion**

**Performance**

We assume that the smaller the summation of alter angle difference, the better visual effect it can achieve. As shown in Table 1, we randomly select some pairs of alters as samples to test the performance of our algorithms. We can easily see that in most cases, the result can be improved significantly when a layout algorithm is applied. The greedy algorithm sometimes falls into a trap of local optimal, which can be even worse than the original layout (i.e. BFS) in a few cases. Nevertheless, its performance is as good as that of Kuhn–Munkres algorithm in most time. Figure 6 provides a specific example to illustrate the difference, with the original layout shown in Figure 2. Although in most cases the original layout always has a larger value of sum of alter angle than greedy and optimal layout, it is preferred by...
users when the scale of ego-networks are tiny for its key advantage of preserving alters’ BFS order.

**Limitations**

Although previous study shows the effectiveness and efficiency of our system, there is still space for further improvement.

Our work is suitable for analyzing the ego-networks whose alter number is not too large (less than hundreds) regardless the scale of an entire network. Thus, it is useful for large-scale network analysis but not for all situations, especially when the size of ego-network is too large. When the number of the common alters of two ego-networks increases to hundreds, the number of links can easily exceed hundreds, and therefore this would lead to a serious problem of visual clutter. To alleviate the visual clutter caused by links, we can use Sankey diagram\(^3^7\) to bundle the lines through the same session into an entire flow. As for the nodes, techniques like heat-map or node collapse is a potential solution. Furthermore, more interactions can be provided to highlight individual node-link-node paths according to users’ demands. We will leave it as our future work.

egoComp is designed for comparing two ego-networks by the use of juxtaposition, so one of the limitations is to compare more than two ego-networks at the same time. For visually comparing multiple ego-networks, we do not need to show so many details as comparing between only two ego-networks. Therefore, it is possible to come up with a new design without such many details in order to compare among multiple ego-networks.

Although the greedy algorithm and Kuhn–Munkres algorithm can decrease the number of crossings of links effectively, they are inevitable to change the original structure of an ego-network, which could result in an altered layout which is less meaningful than the original one (i.e. BFS). To strike a balance between the meaningfulness and cleanliness, our method is to provide different choices of layout algorithms for users.

Although links take much of the display space, they are less informative and only illustrate which alter in LEN corresponds to which alter in REN. In addition, the purpose of using sessions in the middle area is to visually classify alters by their angle attribute to ensure a visually clear view. However, the angle attribute is not the property of the original network data. Thus, we do not take full advantage of the visual space. To convey more informative information by sessions, we can first determine which sessions the links should flow through according to the types of the links (i.e. from hop1 alter to hop1 alter, from hop1 alter to hop2 alter, from hop2 alter to hop1 alter, and from hop2 alter to hop2 alter). Therefore, we plan to redesign the layout algorithms to meet the requirement.

**Conclusion**

In this article, we present egoComp, a visualization technique based on the node-link diagram and juxtaposition method. It aims to facilitate comparing two ego-networks. egoComp allows users to quickly choose two nodes in a graph as egos and visualize the ego-networks expanded by these two egos using a radial layout with hop1 alters in inner semicircle and hop2 alters in outer semicircle. The storyflow-like\(^1^1\) links are applied to illustrate the alters’ position change in left ego-network and right ego-network. To reduce the sum of alter angle difference (equation (1)) and make the view visually clean and symmetric, two optimization algorithms (i.e. greedy algorithm and Kuhn–Munkres algorithm\(^3^5\)) are proposed. By experiments, we find that the performance of greedy algorithm approaches Kuhn–Munkres algorithm in most time. However, in some cases when the BFS order of alters is meaningful (e.g. the alters with sequential order may belong to the same cluster/community), users prefer to use the BFS layout. The usability study demonstrates the effectiveness and usefulness of our system.
Apart from the future work mentioned in section “Discussion,” there are still some other directions to continue this work in the future: first, to facilitate users to choose the right pair of nodes as egos to compare, a new feature of guiding them to select nodes in overview is required. For example, we can provide visual clues for users to inform them what kind of roles these nodes in the overview are playing in. Or for users who have selected one ego and want to choose another one with the most similar ego-network to the selected one, we can also visually inform them by some visual clues. Moreover, in our current design, nodes are distributed uniformly along one-dimensional (1D) arc with specific radius, which may lead to the loss of information about the inner clusters existing in hop1 or hop2 itself. If we position nodes in 2D annulus area rather than 1D arc, it would be helpful to find the inner clusters in hop1 and hop2.

Acknowledgements
The authors wish to thank the anonymous reviewers for their valuable comments.

Funding
This research was supported, in part, by National 973 Program of China (2015CB352503), the Fundamental Research Funds for Central Universities (2016QNA5014), National Natural Science Foundation of China (No. 61502416), the research fund of the Ministry of Education of China (188170-170160502), 100 Talents Program of Zhejiang University, and HK RGC GRF 618313.

References


