3D Visualization of Network Including Nodes with Labels

Hinako Sassa
Ochanomizu University
Tokyo, Japan
hinako@itolab.is.ocha.ac.jp

Takayuki Itoh
Ochanomizu University
Tokyo, Japan
itot@itolab.is.ocha.ac.jp

Mitsuo Yoshida
Toyohashi University of Technology
Aichi, Japan
yoshida@cs.tut.ac.jp

***Abstract*—Visual cluttering is still a severe problem of large-scale network visualization techniques, and therefore many improvements on network visualization have been presented. Meanwhile, many network datasets in our daily life contain label information. We are developing a network visualization technique which overlays a node-link diagram for visualizing the connectivity and a set visualization for representing the label information. Our technique firstly places a set of nodes in an input dataset into a 3D space and provides an interactive mechanism to manipulate viewpoints in the 3D space. The technique then projects the nodes onto a 2D plane and generates a Delaunay triangular mesh connecting the nodes which have the user-specified label on the 2D plane. Finally, it displays the outer boundary of the triangular mesh to represent the region enclosing the nodes which have the label. In addition, supposing that multiple weighted labels can be assigned to the same node, our implementation draws nodes like pie charts to clearly represent the weights of labels assigned to the nodes. This paper shows examples of the visualization applying a co-authorship network dataset.**

***Index Terms*—Network, 3D Visualization, Set Visualization, Label**

I. Introduction

Many network datasets in our daily life contain label information while we may need to suppose that multiple weighted labels are assigned to the same node. For example, social network datasets represent users as nodes, the friendship between users as edges, and community of users as labels. Remark that we need to suppose a user belongs to multiple communities because communities focus on business, hobby, regional information, and other various information. Here, it is quite interesting if we can visualize the relationship between the connectivity of users and their belonging communities. Addressing this issue, we are developing a new visualization technique combining a 3D network layout and a 2D set visualization so that we can visualize the relationship between the community and friendship.

There have been many existing techniques which extracts communities based on the connectivity to visualize communities on the network datasets. We are addressing a different problem supposing community information is provided as labels independent from connectivity information. Also, there have been many existing techniques to display labels assigned to the nodes of the input network datasets; however, many of the studies suppose that just one label is exclusively assigned to nodes of the network. Our study is not based on this assumption: the study supposes that multiple labels may be assigned to a node.

In addition, visual cluttering caused by overlapping nodes and crossing edges is still a severe problem of large-scale network visualization Various network layout techniques including force-directed algorithms and dimension reduction schemes have been discussed to realize layout avoiding overlapping nodes and crossing edges in the past years. However, poor comprehensibility may be caused while placing the nodes onto a 2D space even applying the above techniques.

This paper proposes a technique combining the following two types of visualization addressing the above-mentioned problems.

- 3D network visualization with interactive viewpoint operation.
- 2D set visualization to represent labels of nodes.

This technique constructs sets from labels assigned to the nodes independently from their connectivity and overlays the set visualization result on the network visualization. This representation makes users easy to understand the relationship between the connectivity of the network and the distribution of labels.

Moreover, we suppose that multiple labels can be assigned to a node where the labels can also be weighted with real values. In other words, weights of labels of nodes are represented as m-dimensional vectors supposing there are m labels in a given dataset. Our current implementation draws the weights of labels of a node like a pie chart. Also, this technique displays polygons enclosing a set of nodes which the user-specified labels are assigned overlaying with the network visualization. Our implementation generates the polygons from Delaunay triangular meshes which connect corresponding nodes which are projected onto a 2D space. The technique draws the outer boundary of the triangular mesh to represent the set of nodes. This implementation can represent regions where the nodes with user-specified labels are concentrated.

The remainder of this paper is as follows. Section 2 introduces the related work. Section 3 describes the proposed technique followed by Section 4 introducing our experiments and discussing the effectiveness of our technique. Finally, Section 5 concludes this paper and discusses our future work.

II. Related Work

A. Network Visualization with Attributed Nodes

There have been several techniques on network visualization with attributed nodes, where the attributes include labels and feature vectors.

Itoh et al. [2] proposed a network visualization method where labels assigned to the nodes. The method applies a graph layout algorithm taking into account the combination of connectivity and labels. Many existing graph clustering techniques had a problem that important nodes connected to many edges are hidden inside large clusters. On the other hand, this method solved the problem by applying a clustering method based on commonality of connected nodes so that important nodes are separated from large clusters. Moreover, this method can take into account the feature vectors of nodes while applying the clustering method.

Itoh et al. [3] also presented another network visualization method which applies a hybrid force-directed and space-filling layout algorithm. This method places a group of nodes which have the same items close to each other after applying a hierarchical clustering algorithm. This method has a problem that its complexity increases rapidly with the increase of the number of items. Meanwhile, this method represents items associated with a node as fans so that the node looks like a pie chart as well as our technique presented in this paper. It divides a circle into the fixed number of equally shaped fans: it does not represent the weights of items.

B. Network Visualization on 2D and 3D

While a huge number of studies on 2D network visualization techniques have been presented, there have also been studies on 3D network visualization techniques. Several studies experimentally compared comprehensibility of 2D and 3D network visualizations. For example, Ware et al. [4] presented an experiment which verifies comprehensibility from the time which was taken to read network and the ratio of error in reading it. As a result, this paper concluded that 3D visualization was preferable to quickly read the visualization results comparing with 2D visualization if the number of nodes is between 100 and 140. Also, the ratio of reading error of participants was at most 20% while using 3D visualization against it was as most 40% while using 2D visualization. Furthermore, Ware et al. verified the rotation operations of 3D visualization. They compared eight implementations: four variations of rotation operations including no rotation with 2D visualization, automatic rotation, manual operation, and synchronization with head movement, displayed by a desktop display and stereo head mount display. Participants required the shortest time with manual rotations without using the stereo head mount display. We respected this result and applied the manual rotation operation in our implementation.

C. Set Visualization

Set Visualization techniques represent sets of nodes. Bubble Sets [5] and Line Sets [6] are typical famous 2D set visualization techniques.

Bubble Sets draws the sets by closed contours surrounding corresponding nodes and dodging other nodes. Line Sets draws the sets by connecting corresponding nodes by smooth curves. These techniques can be applied with a variety of layouts of nodes, and also overlay on various visualizations including scatterplots, trees, networks, and maps. On the other hand, poor comprehensibility may be caused if the number of corresponding nodes increases and therefore shapes of sets get complicated. In addition, these techniques may require large computation time with a large number of nodes, and therefore it may be difficult to apply to interactive visualization systems.

We discussed to represent sets by simpler shapes of objects to reduce the computation time and realize more interactive visualization, and represented sets as convex hulls as a trial [1]. Convex hulls are the smallest convex polygons surrounding all target nodes. There have been many algorithms generating convex hulls, including gift-wrapping and Quick Hull [7] algorithms. These complexities are smaller than those of Bubble Sets and Line Sets. However, we concluded that the convex hull is not appropriate for set visualization in many cases because the convex hull often draws unnecessarily large polygons surrounding many non-target nodes.

Watanabe et al. [8] applied Delaunay triangulation to evaluate shapes of point clouds drawn in scatterplots. The Delaunay triangulation algorithm connects given sets of points to generate triangular meshes. Watanabe et al. generated triangular meshes from all points drawn on scatterplots, and calculated metrics of the appearance of scatterplots after removing unnecessary long edges of the triangular meshes. The technique applies a similar algorithm to generate polygons representing the sets from Delaunay triangular meshes.

III. Proposed technique

This section presents the processing flow of the proposed technique. Figure 1 illustrates the processing flow of this technique. First, the technique places nodes in a 3D space and specified the viewpoint position with interactive operations. Next, it projects the nodes onto a 2D space and overlays the set visualization. The technique modifies the polygons representing the sets of labels whenever a user specifies the viewpoint position.

Section 3.1 describes the 3D network visualization, followed by Section 3.2 describing the implementation detail of set visualization.

A. Network visualization in a 3D space

This technique firstly calculates positions of nodes in a 3D space. The technique applies a node clustering algorithm presented by Itoh et al. [2] which separates important nodes connected with a large number of other nodes from large clusters. Next, it calculates distances between arbitrary pairs of clusters from the number of edges connecting the pairs and similarity of weights of labels. Finally, the technique forms a distance matrix and calculates the positions of clusters applying Multi-Dimensional Scaling (MDS).

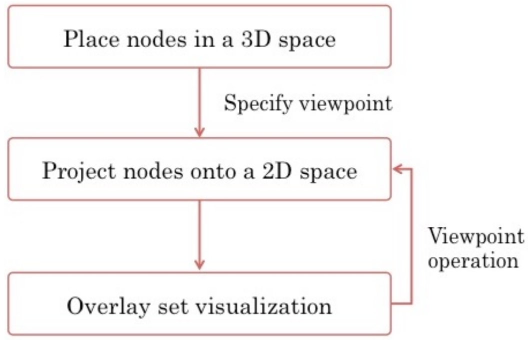

Fig. 1. Processing flow

It is often difficult to understand the overall structures of given network datasets if we draw every edges in the datasets. Our implementation can limit the drawing of edges to ones connecting to the user-specified node to avoid this situation.

Our implementation assigns individual colors to each of labels, and paints nodes with the color corresponding to the most weighted label. Or, it draws a node as a set of fan-shaped colored regions like a pie chart to represent the weights of all labels.

We argued that 3D visualization is better in reading time and error comparing with 2D visualization as Ware et al. presented [4] in Section 2. In addition, we preferred to implement the 3D network visualization because we can freely operate the viewpoint and generate different views without re-layout. It is especially helpful in this study: we can change the viewpoint to obtain better views when a visualization result does not bring a comprehensive view of user-interested labels.

B. Set Visualization

Subsequently, this method generates a polygon enclosing a set of nodes which a particular label is assigned by applying Delaunay triangular method, and overlays on the visualized network. First, it generates a triangular mesh connecting extracted set of nodes which a particular label is assigned. This process is applied on a 2D screen coordinate system. Next, it deletes edges of the mesh whose lengths are longer than the user-specified threshold. At the same time, it deletes the triangles enclosed by the deleted edges. After this operation, it renders the set of remaining triangles that correspond to the set by drawing the region boundary of the remaining triangles. Then, it overlays circles on the nodes isolated by the edge deletion operations. Our previous method applying convex hulls [1] to represent the set had a problem that it enclosed many non-set nodes because it generates unnecessarily large polygons. The method presented in this paper solved this problem by applying the Delaunay triangular mesh.

Furthermore, our method supposes to control the viewpoint manually. Our method reconstructs the polygon enclosing the target nodes whenever the viewpoint is moved. Therefore, it needs to quickly generate the polygon for this interactive operation. Our method can freely rotate large-scale networks containing thousands of nodes which reconstructing the polygon because the complexity of Delaunay triangular mesh is $O(\log n)$ against n nodes.

When our implementation draws all polygons corresponds to all labels at the same time, the visualization results get less comprehensive because of the overlap of multiple polygons. Therefore, our method draws only user-specified polygons to remain the comprehensibility.

IV. Experiments

We implemented this method with Java Development Kit (JDK) 1.8. Especially, we implemented the 3D network visualization by extending the implementation of Itoh et al. [2]

A. Network of co-authorship

This paper describes the example dataset based on the publication bibliography of NERC Biomolecular Analysis Facility (NBAF). This dataset describes a co-authorship network consists of 1821 author as nodes and 11097 co-authorship as edges. Itoh et al. [2] specified the 11 words frequently appeared in the titles of papers and calculated 11-dimensional feature vectors for each node from the frequency of these words. Moreover, they colored each node based on words corresponds to the dimension which has the maximum value among the 11 values. This experiment defines the labels as a set of words corresponds to dimensions which have values larger than the user-specified threshold.

Color	Label
	Genetic
	Molecular
	Loci
	Microsatellites
	Isolation
	Inbreeding
	Transcriptions
	Expression
	Bacterial
	Breeding
	Polymorphic

Fig. 2. The correspondence table of each node and label

Figure 2 shows the correpondences between colors and words. The red is Genetic, the orange is Molecular, the yellow is Loci, the yellow green is Microsatellites, the green is Isolation, the blue green is Inbreeding, the sky blue is Transcriptions, the blue is Expression, the indigo is Bacterial, the purple is Breeding, and the pink is Polymorphic.

Figure 3 shows an example of network visualization results before rendering a set of labels. Figure 4 is an example where

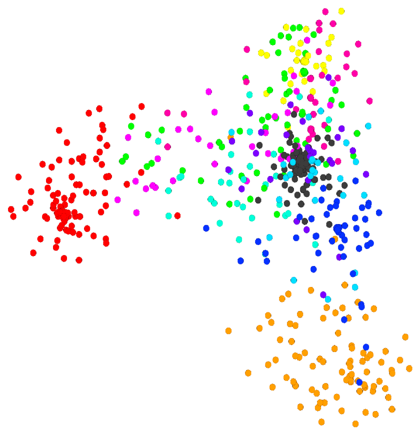

Fig. 3. The network before rendering label

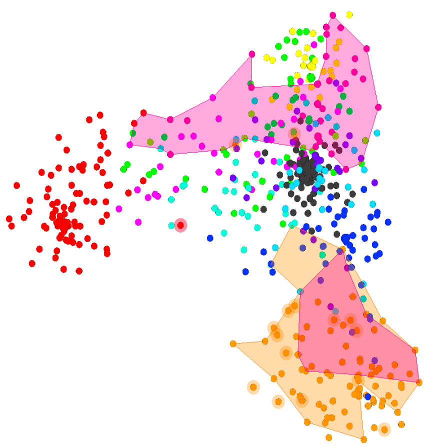

Fig. 4. The network after rendering label

a set of plygons corresponding to the labels are overlaid on the network visualization shown in Figure 3. In addition to, the nodes on circumference of polygon is included in the set.

Here, we need to distinguish the labels of nodes only by their colors if we have the representation as shown in Figure 3. It is especially difficult to distinguish colors of small nodes if the hues of colors are near, such as yellowish green and green, blueish green and sky blue, and purple and pink. Our representation in Figure 4 is easier to distinguish. However, it can focus on the distribution of the orange and the pink compared to figure3 with set visualization of our method like figure4. Additionally, it can guess that these two labels have deep relationship because of overlap these sets.

Figure 5 is another example which renders the nodes like pie chart according to the weights of labels. Here, gray nodes are not assigned to any of labels It shows which node is given what label tough it shows multiple selected labels with this function. For example, Figure 5 shows that the nodes circled in pink have both pink and orange labels represented as pie

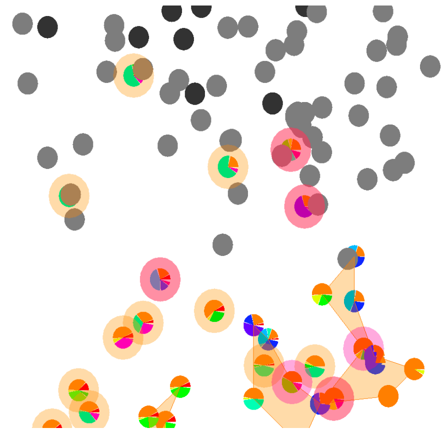

Fig. 5. The nodes drown like pie chart

charts. Most of the nodes have larger orange fans and smaller pink fans. It suggests that the word "Molecular" corresponds to orange is more frequent than the word "Polymorphic" corresponds to the pink in the papers of authors corresponding to these nodes.

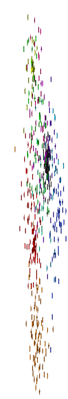

Fig. 6. Figure rotated Figure3

Figure 6 shows a visualization result from a different viewpoint controlled by the rotation operation. This operation makes easier to grasp the depth of the 3D space. On the other hand, the layout of network was thin in this experiment; this result made the rotation operation less effective. We would like to replace the 3D network layout algorithm as a future issue.

B. Visualization of a retweet network of Japanese political parties on Twitter

We visualized datasets of networks of Twitter users connected by retweets of tweets by official accounts of Japanese political parties. We estimated that this visualization result would show the behaviors of retweets of political parties: some users might retweet tweets of a specific party, or others might retweet tweets of various parties. We can analyze what kinds of users are following those who have retweeted tweets of a specific party by this visualization. Twitter datasets are easy to construct for us because we can get various information via Web API easily comparing with other social networking services. We define the users as nodes, relationships between users as edges, and political parties retweeted by the users as labels. We constructed 13-dimensional feature vectors from the frequency of retweets for each political party. Figure 7 shows the correspondence between colors and political parties. This section introduces visualization results with two datasets (called dataset A and B) described in the same data structure.

Color	Label
	The Happiness Realization Party (HPP)
	Liberal Democratic Party of Japan (LDP)
	Komei
	New Party DAICHI (NPD)
	Liberal Party
	Social Democratic Party (SDP)
	Japanese Communist Party (JCP)
	The Party for Japanese Kokoro
	The Assembly to Energize Japan
	Japan Innovation Party
	The Democratic Party (DP)
	Party of Hope
	The Constitutional Democratic Party of Japan (CDP)

Fig. 7. the correspondence table of each node and label

Figures 8 9 are the result with the labels of the "LDP", "Komeito" and "The Party of Hope." Distributions of these parties in both datasets are very similar. Therefore, we estimate that the users interested in the LDP tend to be also interested in Komeito and the Party of Hope. Actually, this result is expected because LDP and Komeito are the ruling parties.

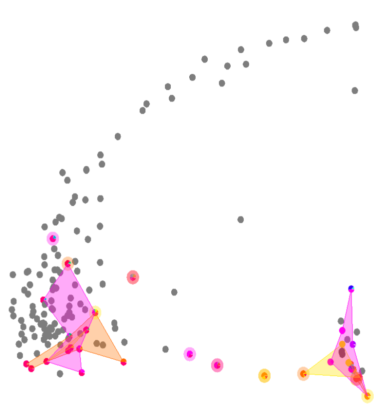

Fig. 8. The network of dataset A with sets of LDP, Komeito and The Party of Hope

Figures 10 and 11 are further results with the labels of the "CDP" and "JCP" as well. These figures also demonstrate the distributions in both datasets are similar as well as the former visualization result. Therefore, we estimate the users

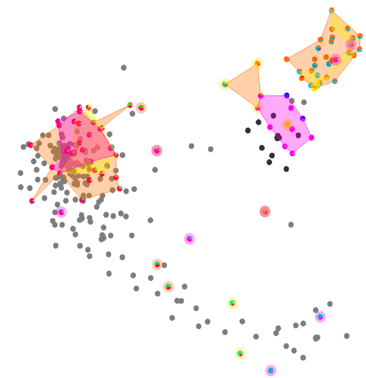

Fig. 9. The network dataset B with sets of LDP, Komeito and The Party of Hope

who retweeted JCP are also likely to be interested in the CDP. Additionally, the latter result shows more interest in more users because the labels in the latter result are more widely spread against the former result. Especially, all the nodes have larger pink fans depicting larger weights. We estimate the reason for this result is that CDP was a new political party formed in 2017 and therefore was attracting users' attention.

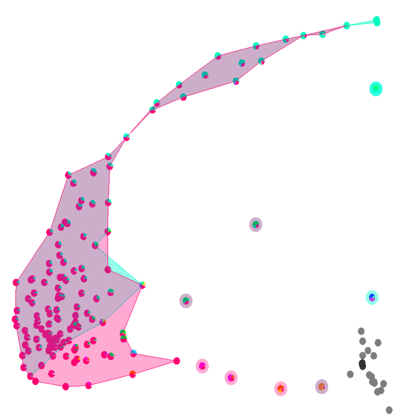

Fig. 10. The network of dataset A with sets of Communist party, and CDP

V. Conclusion and Future Work

We proposed a network visualization technique which calculates the layout in a 3D space and overlays polygons enclosing a set of nodes which user-specified labels are assigned. This paper also introduced two case studies with this technique with paper co-authorship and retweet network datasets.

We have a problem of time-consuming operations for finding the best viewpoint which brings the most effective visualization results. We would like to implement an automatic

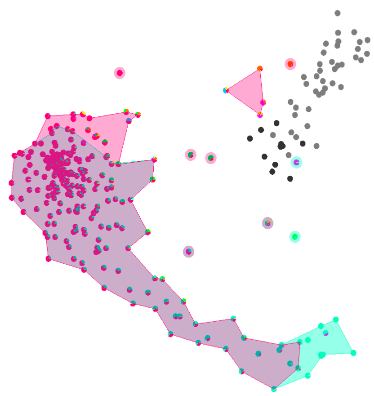

Fig. 11. The network dataset B with sets of Communist party, and CDP

viewpoint recommendation mechanism to resolve this problem. As a metric measuring the effect of visualization, we will challenge to define an evaluation function regarding shapes of polygons representing the sets. By evaluating the polygons from multiple viewpoints, we will realize the visualization from the viewpoint automatically.

Another problem is that interactive operations with our current implementation are feasible to up to several thousands of nodes. One reason for this limitation is that it takes longer computation time for projection of large-scale datasets. As a solution to this problem, we would like to project networks in advance from several highly evaluated viewpoints.

VI. Acknowledgments

This work has been partially supported by Japan Society of the Promotion of Science under Grant-in-Aid fir Scientific Research.

References

[1] 佐々, 伊藤, ネットワークの 3 次元可視化と集合可視化の重ね描きの一手法, 第 10 回データ工学と情報マネジメントに関するフォーラム (DEIM), E5-4, 2018.
[2] T. Itoh, K. Klein, "Key-node-Separated Graph Clustering and Layout for Human Relationship Graph Visualization," IEEE Computer Graphics and Applications, Vol. 35, No. 6, pp. 30-40, 2015.
[3] T. Itoh, C. Muelder, K. Ma, J. Sese, "力学モデルと空間充填モデルの併用によるアイテム集合付きネットワークの可視化手法," 研究報告グラフィクスと CAD（CG）, Vol.137, No. 14, pp. 1 - 6, 2009.
[4] C. Ware and G. Franck, "Evaluating Stereo and Motion Cues for Visualizing Information Nets in Three Dimensions," ACM Transactions on Graphics, Vol. 15, No. 2, pp. 121-140, 1996.
[5] C. Collins, G. Penn, S. Carpendale, "Bubble Sets: Revealing Set Relations with Isocontours over Existing Visualizations," IEEE Transactions on Visualization and Computer Graphics, Vol. 15, No. 6, pp. 1009 - 1016, November/December 2009.
[6] B. Alper, N. H. Riche, G. Ramos, M. Czerwinski, "Design Study of LineSets, a Novel Set Visualization Technique," IEEE Transactions on Visualization and Computer Graphics, Vol. 17, No. 12, pp. 2259 - 2267, December 2011.
[7] C. B. Barber, D. P. Dobkin, H. Huhdanpaa, "The Quickhull Algorithm for Convex Hulls," ACM Transactions on Mathematical Software, Vol. 22, No. 4, pp. 78 - 83, 1996.
[8] A. Watanabe, T. Itoh, M. Kanazaki, K. Chiba, "A Scatterplots Selection Technique for Multi-Dimensional Data Visualization Combining with Parallel Coordinate Plots," International Conference Information Visualisation (IV), Vol. 21, pp. 469-483, 2017.