Journal of Visualization

Visualization of Bipartite Relations between Graphs and Sets

--Manuscript Draft--

Manuscript Number:	JOVI-D-14-00112
Full Title:	Visualization of Bipartite Relations between Graphs and Sets
Article Type:	Regular Paper
Abstract:	In many application domains, we encounter data which involves a graph encoding certain relationships and a set of items related to the graph. One example is in social websites where the users interact with each other, and share their interests on different items such as music or books. In this case, the direct interactions among the users can be represented as a graph, and the items like music or books can be represented as a set. People are often interested in the bipartite relation between the graph and the set. They might want to know the similarity or difference of the items liked by themselves and by their friends. In this paper, we propose a visualization framework designed for the exploration and analysis of relations involving a graph and a set. Our system consists of two major components: an enhanced graph view and a radial view. The enhanced graph view shows a social network of people and statistical information about people's interested items; and the radial view is designed to show people's interests, the overlapping of their interests, and recommended items based on their interests. The combined use of the two visualization components can facilitate the discovery of various relational patterns underlying the links connecting the graph and the set. The experiment on the real dataset demonstrates the effectiveness of our technique.

Noname manuscript No.
(will be inserted by the editor)

Visualization of Bipartite Relations between Graphs and Sets

Hong Zhou · **Panpan Xu** · **Huamin Qu**

Received: date / Accepted: date

Abstract In many application domains, we encounter data which involves a graph encoding certain relationships and a set of items related to the graph. One example is in social websites where the users interact with each other, and share their interests on different items such as music or books. In this case, the direct interactions among the users can be represented as a graph, and the items like music or books can be represented as a set. People are often interested in the bipartite relation between the graph and the set. They might want to know the similarity or difference of the items liked by themselves and by their friends. In this paper, we propose a visualization framework designed for the exploration and analysis of relations involving a graph and a set. Our system consists of two major components: an enhanced graph view and a radial view. The enhanced graph view shows a social network of people and statistical information about people's interested items; and the radial view is designed to show people's interests, the overlapping of their interests, and recommended items based on their interests. The combined use of the two visualization components can facilitate the discovery of various relational patterns underlying the links connecting the graph and the set. The experiment on the real dataset demonstrates the effectiveness of our technique.

Keywords Bipartite relations · Graph visualization · Radial visualization · Bar charts · Information visualization

1 Introduction

Many applications involve a social network of people and a set of items related to those people. For example, friends in the Facebook share their favorite music or books. Friends

Hong Zhou
Shenzhen University, Guangdong, China. E-mail: hzhou@szu.edu.cn

Panpan Xu
The Hong Kong University of Science and Technology, Hong Kong, China.
E-mail: pxu@cse.ust.hk

Huamin Qu
The Hong Kong University of Science and Technology, Hong Kong, China.
E-mail: huamin@cse.ust.hk

(personal social network) or colleagues (professional social network) may follow different topics in Twitter. Researchers form collaboration networks by co-authoring papers but they often have different research interests.

In these applications, people are often interested in the bipartite relationships between the social network and the set of items (e.g., books, topics, interests). One possible scenario is that people often want to know the interests of their friends, how diverse their interests are, and what kind of interests they possibly share. People may explore their social network and ask questions like *"who have similar interests with me but are not my close friends yet?" "who have much broad interests than me?"* *"which friends have quite different interests than me?"* Or they can start from the items in the set, select some items, and ask *"who are interested in these items and what are their relationships"*. They may also want some suggestions for similar items based on what they already have.

To answer these questions, we need to know three different kinds of relationships: the social relation between people, the set relation between people's interested items (e.g., books, topics, interests), and the association relation between the people in the social network and the items in the set.

There are well established methods to visualize any one of the relationships. For example, we have layout methods for network visualization, clustering methods for items in a set, set relationship visualizations like Venn diagram, and bipartite graph visualizations. However, there still lack effective visualizations to reveal all the three kinds of relationships simultaneously. It is very desirable to develop a comprehensive framework which can support bi-directional explorations of bipartite relationships between a social network and a set of items. Users can start from the social network and highlight persons they want to explore, and then the set items of selected persons and their relationships (e.g., superset, overlapping) will be revealed. After that, based on those set items, the system will automatically suggest some similar items for users. On the other side, users can choose some items and then system will highlight people who are interested in these items and their social relationships in the graph.

In this paper, we present a comprehensive visualization system to help users explore the bipartite relations between a graph and a set. Our system consists of two components: a) a graph view with enhanced bar charts to show the social network and statistics about people's interests; b) a radial view to show the items interested by some selected persons, the set relationships of their interests, and also some recommended items. We demonstrate a typical usage scenario for our system: visual exploration and recommendation of music artists based on the artist similarity, the user's profile, and the user's friends.

The major contributions of this paper are as follows:

- A graph view enhanced with bar charts to show a social network and statistical information about their interest distributions for people in the social network.
- A radial visualization design to show people's interests, the overlapping of their interests, and recommended items based on their interests.
- A visualization system which seamlessly integrates the above components to facilitate the visual exploration of the bipartite relations between people's social network and a set of their interested items.

2 Related Work

Our work draws on research in several categories. In related works, we first review the current existing visualization techniques for general graphs and bipartite graphs. Then we discuss some recent research works on visualizing heterogenous networks. Finally, we present related set visualization methods.

2.1 Graph visualization

Two commonly used visual representations for *general graphs* are the node-link diagram and the adjacency matrix [12]. The two can also be combined by using linked views [14] or partial matrix and partial node-link representations [15]. One of the major problems in drawing node-link diagrams is assigning coordinates to the nodes, and many graph layout algorithms have been proposed [2]. For graphs having nodes with domain specific attributes, the nodes can be arranged according to the attributes values [31]. For adjacency matrices, graph structures like clusters can be made more evident by permuting the matrix rows and columns [14].

Bipartite graph, which represents relations between entities in two disjoint sets, can be found in many application domains. Similar to the general graphs, bipartite graphs can be drawn as node-link diagrams or matrices. For node-link diagrams, one common practice is to assign disjoint drawing space to ensure visual separation of the two sets of nodes, and arrange the nodes within the spaces to achieve the aesthetic goal of minimizing edge crossings or edge lengths. For example, the nodes can be placed on two parallel axis [23], or two concentric spheres [22]. Another drawing style for bipartite graphs restricts the nodes in one set to fixed positions (hence "anchors") and treating nodes in another set as "free nodes" which can be placed using force-directed model [20] [28].

Bipartite graph visualizations have been used for showing relations between keywords of publications and the corresponding years of publication to show the topic shifts overtime [28], or show the relationship between words and other entities like research teams or documents [22] in order to make the research literature more understandable.

Our work is inspired by the aforementioned anchored layout technique, but since we deal with a graph and a set and the analytical tasks are different, the above mentioned techniques cannot be directly used.

2.2 Heterogenous network visualization

A lot of researches have been done in recent years for the visual exploration and analysis of *heterogenous networks*. Heterogenous network are composed of many different types of entities and relations. NetLens [17] and PaperLens [18] supports visual explorations of networks that fits a "content-actor" data model (such as scientific publications and authors), where both the content and actors consists of networked data. Instead of using nodelink graphs to represent the relations, these two systems use multiple coordinated views of lists, histogram overviews to help users explore the dataset. EdgeMap [8] displays the implict and explicit relations among a set of data items, with the former encoded by spatial positions and the latter encoded by drawing links. Visual analysis systems proposed in [13] [19] support interactive modeling of graphs from relational data tables. OntoVis [25] uses an ontology graph of node types to guide the filtering and abstraction of a large, heterogenous graph. FacetAtlas [5], JigSaw [27], Solarmap [4], TextWheel [7] extract different types of entities and their relations from text corpus and visualize them.

The data model used in our paper involves heterogenous relations. In particular, we further extend the ”content-actor” model with set relationship visualization and similar item recommendations to facilitate users to explore the bipartite relationships between a graph and a set.

2.3 Set Relationship Visualization

The representations of set relations have been studied and received attentions from visualization researchers. Freiler et al. [9] proposed a new method to visualize set-typed data. Their method is especially effective for high dimensional data. Simonetto et al. [26] designed Euler-like diagrams to visualize overlapping sets. Collins et al. [6] developed bubble-like shapes to enclose items belong to the same set into bublles. Instead of using bubbles, Alper et al. [1] adopted smooth curves to connect items in the same set. Our work also needs to reveal the set relations of people according to their interested items. However, we also need to provide other information like item similarity. In this paper, we propose a novel radial layout which can reveal item similarity, people and their interested items, and set relations of people according to their interested items.

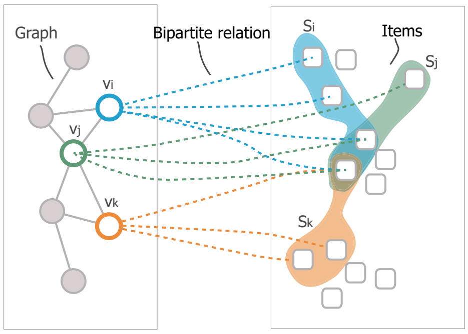


Fig. 1 Our data model consists of a graph G and a set S. There exist bipartite relations between the graph and the set as the items in the set S may belong to the nodes in the graph G.

3 Data Model and System Overview

In this section, we first introduce our data model. Then we discuss analytical tasks and some design considerations. Finally we give an overview of our system.

3.1 Data Model

We first give a description of the data model used in our paper. It can be formulated as follows: Let $G = \{V, E\}$ represents a graph and $S = \{s_1, s_2, ...s_n\}$ represents a set. The bipartite relation exists between G and S as there is relation between V and S. Given the bipartite relation, each $v_i \in V$ corresponds to a subset $S_i = \{s_{i1}, s_{i2}, ...s_{ij}\} \subseteq S$, which contains its related items. There is also similarity defined for entities within S. Fig. 1 illustrates our model. The data model can be applied such that G represents the social graph and S refers to items such as music, books, research interests.

3.2 Visualization Design

Our system is designed to visualize the above mentioned data model and we want to achieve the following goals:

Our system should show the social relationship between people and also key statistics about their interests. To achieve this goal, we use traditional graph visualization and enhance its nodes with bar charts to encode the statistics like the number of interested items each person has.

Our system should reveal the set relationships among people according to their interests. The system should allow users to intuitively compare the interests among a set of selected people, and interests between these people and all other people. To achieve this goal, we introduce a radial view of the set relations among multiple persons, and a graph view with bar charts to compare the interests of a subset of people with all other people.

The system should display the items interested by people based on similarity and also make recommendations of similar items based on items they already have. We use a force-directed layout for the items based on their similarity to the items that people already have.

The system should allow bi-directional exploration of the relationships between a graph and a set. To achieve this goal, we adopt a linked view with one window displaying the graph and another window showing the set. If users select some people in the graph window, the set of items they are interested in will be revealed in another window. Similarly, if users choose a few items in the set window, people who are interested in these items will be highlighted in the social network window. The color is used to establish the correspondence between the graph nodes and the items.

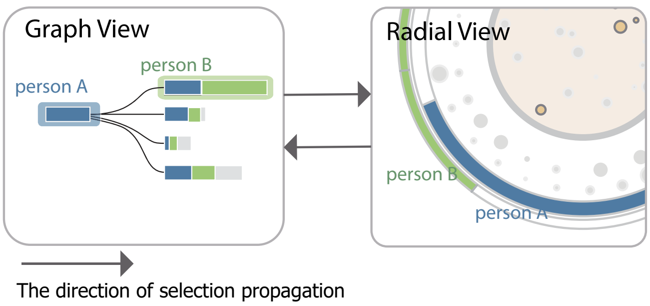


Fig. 2 System overview. Our system consists of two major components: (a) a graph view to show the social network. The social network is enhanced with a bar chart for each node to show the number of interested items for this node and their shared interests with a few chosen people. (b) a radial layout to show the items interested by a few chosen people, their set relations, and also some suggested items.

3.3 System overview

Fig. 2 shows the overview of our system. Our system adopts a linked view design and there are two major components in our system: 1) A graph view which shows nodes and their relationships indicated by edges, and statistic information represented by barcharts related

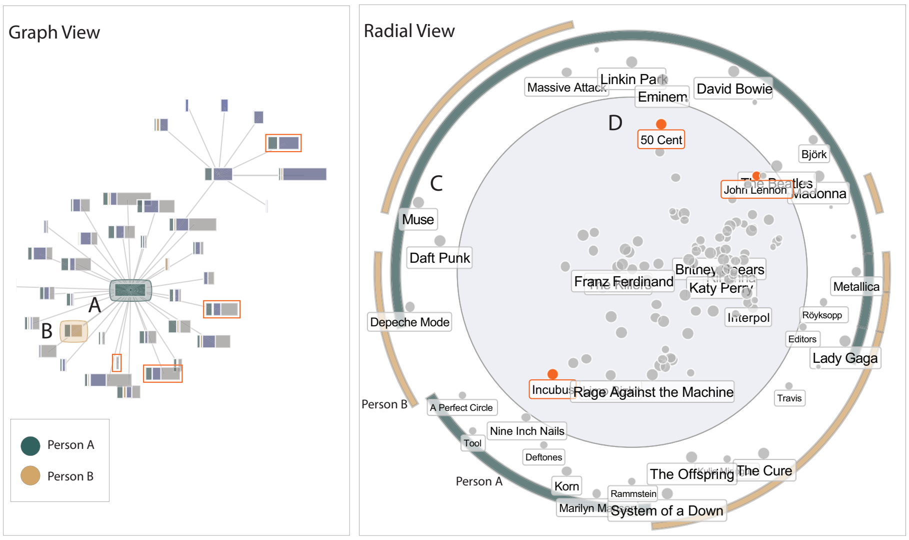


Fig. 3 A visualization example of bipartite relations between a social network of friends (i.e., left graph view) and a set of music artists (i.e., right radial view) they are interested in.

to their interests; 2) A radial view to show the set memberships of the items and the recommended items. Users can start from the graph and pick up a few nodes. Each node will be highlighted by a distinct color. Then all the items related to the nodes will be displayed in the radial layout and some similar nodes will also be suggested and displayed. Each arc surrounding the radial layout represents a node or a node group in the graph and color is used to establish the correspondence between the arc in the radial view and the node or node group in the graph view. Moreover, the barcharts (i.e., statistical information) related to all the nodes in the graph will be updated. Based on the statistical information, users can explore the social network, and select more persons for further investigation. If users are interested in the suggested items in the radial view, they can select a few interested items, and then in the graph view, people who are interested in these items will be highlighted.

Fig. 3 illustrates a more concrete example. This example shows the bipartite relations between a social network of friends and a set of music artists they are interested in. Two friends are chosen in the left graph view and highlighted with peacock blue (person A) and chrome yellow (person B). Their interested items are shown in the ring region (C) between the outer arcs and the inner circle in the right radial view. The items in the inner circle (D) are those recommended by the system given the interests of the two persons. Users can select some items (highlighted in orange) in the right radial view, and persons who are interested in these items are highlighted accordingly in the left graph view. The barchart in each node of the graph view encodes the amount of interest overlap with the selected persons.

4 Visualization Schemes

In this section, we will introduce the visual design of each major component in our system.

4.1 Enhanced graph view

The social network is represented as a graph. Our system supports two kinds of layout: a graph layout to show a group of nodes and their relations, and a tree layout to show the social network of a user selected node. The tree layout is basically a breadth-search tree for the selected node which encodes the distance of all other nodes to this node. As the goal of our system is to explore the bipartite relation between the social network and the item set, we further enhance the graph view by encoding some statistical information related to the items interested by each person. For each node, we provide a bar chart to encode the following information: a) The length of each bar encodes the total number of items interested by this node. b) If users select a few nodes by using different colors, then the bar chart will be updated accordingly by showing the overlapping of the interest with those selected nodes. Fig. 4 illustrates our enhanced graph view.

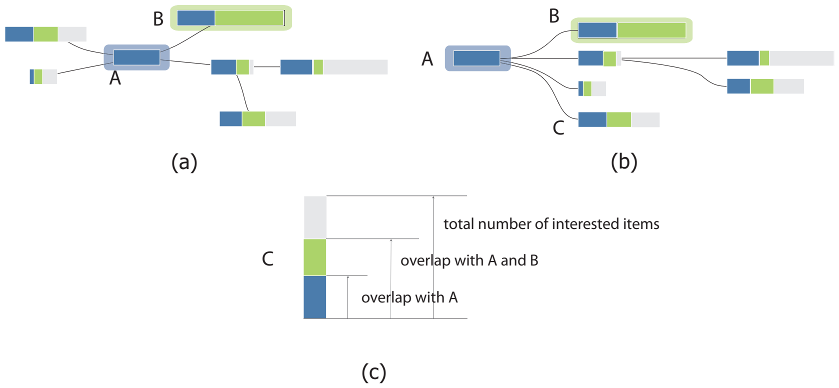


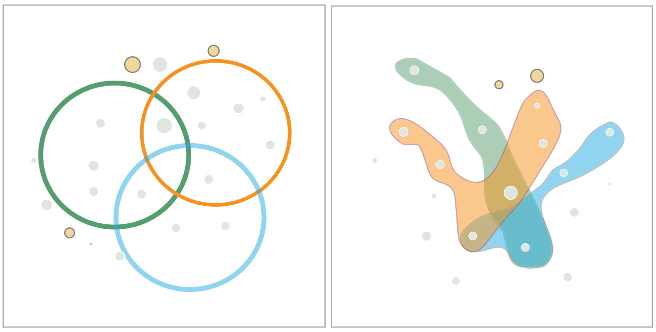
Fig. 4 Our graph view supports two layouts: (a) a traditional graph view shows a social network; (b) a tree view for a chosen node. The view is a breadth-first search tree which presents the nodes according to their distance to the chosen node. (c) The graph is enhanced with a bar chart for each node to encode the number of interested items for this node and the distribution of the items compared with a few highlighted nodes.

4.2 Radial View

4.2.1 Design Rationale

The radial view is designed to provide a graphical display of three types of relations: the similarity relations between the items in the set, the association relation between the node in the graph and the items in the set, and the set relations of people's interests. Specifically, we want to achieve three goals for this view: 1) By default, all items in the set should be displayed based on their similarity and similar items should appear together. 2) If users select some people or group in the social network, the display should be updated and the items interested by these people or groups should be highlighted. Meanwhile, their set relationships should be revealed. 3) We also want to recommend some items which are similar or relevant to the items interested by the selected groups. Recommendation is a highly useful feature in social media. If users can select some items in the set, the system will highlight people in the social network who are interested in these items.

We find it is a challenging task to achieve these three goals. For the first goal, there exist well established methods like MDS to layout items based on their similarity. For the second goal, there are various visualization schemes for set relations like Euler or Venn Diagram, Hypergraph, Bubbleset [6], and Linesets [1]. For the third goal, there are also efficient recommendation methods available to find similar or relevant items. However, there is no trivial solution to an integrated visualization which can achieve these three goals simultaneously. For example, Venn diagram, Bubbleset, and Linesets [1] are good at showing the set relations. But we also need to layout suggested items which do not belong to any set but they should be positioned according to their similarity to the items in the selected set. Fig. 5 illustrates some drawbacks of Venn Diagram and Bubble Sets for revealing set relationships and node similarities.



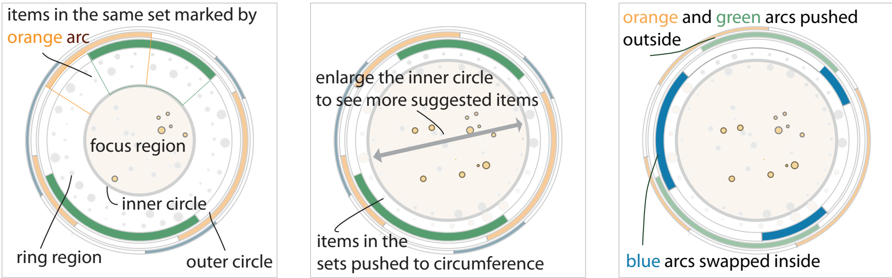
(a) (b)

Fig. 5 Venn Diagram and Bubble Set for set relation visualization: (a) Venn diagram. It is difficult to position suggested items. (b) BubbleSet. It is also not easy to position suggested items. It is also challenging for both methods to provide focus+context view for interested items and suggested items.

We further use Linesets [1] as an example to illustrate several challenges we face. The Linesets method uses lines connecting the node subsets to depict the set memberships and overlappings in a graph. However, the method cannot easily answer some queries: 1) Find nodes in the graph that are closely linked to nodes in one or more sets. For example, we may want to find items that are similar to those already in a person's profile. 2) The relatedness of two sets. This is different from set overlapping. For example, two people may have disjoint sets of interests or their interests only have a small amount of overlap, but the interests in their profiles are highly related to each other. 3) The clusters of interests within a person's profile. For example, a person's music collection may have two clusters of similar artists, such as country music artists and rock bands. Next we will introduce our design to tackle these challenges.

4.2.2 Encoding Scheme

After investigating various designs, we come up with a radial layout design which can achieve all our goals and meanwhile offer great flexibility for user interactions. Our design is inspired by the idea of "anchoring" data items on a radial circumference as references and placing other items within. This idea can be found in the anchored layout of bipartite graphs [20] [21] and visualization of multidimensional datasets where the dimensions act



(a) (b) (c)

Fig. 6 The radial layout: (a) The encoding scheme of the radial layout. Our design consists of three components: an inner circle, an outer circle, and arcs. The arcs represent the people highlighted in the social network. The overlaps of arcs encode the set relations. The items interested by the chosen people will be positioned in the ring region between the inner circle and outer circle. The suggested items will be put inside the inner circle. (b) Focus+context view can be easily achieved by adjusting the size of inner circle. A smaller inner circle will give the focus view to the interested item set while a larger inner circle will allow users to focus on the suggested items. (c) The arcs can also be swapped to bring in different user groups for comparisons. The design allows best comparisons for the inside 2 or 3 arcs. Users can swap the arcs to bring their interested groups inside for better comparisons.

as anchors [24] [16]. Our radial visualization design consists of three components: an inner circle, an outer circle, and arcs. Fig. 6 shows our encoding scheme. Next we will introduce each component.

Arcs The arcs represent different people or groups. We use color to establish the correspondence between the radial view and the graph view. For example, if users select three persons and each person is assigned a unique color for labeling (e.g., red, green, and blue), then three layers of arcs with these colors will appear in the radial view and represent different persons.

Ring Region The region between the inner circle and outer circle is called the ring region which is used to layout items interested by the selected groups. The ring region will be divided into different sectors according to the outside arcs and there is a one-on-one correspondence between the items in the sectors and the arcs representing people (Fig. 6). If there is only one arc outside the sector, then all items in this sector are only interested by the person represented by this arc. If there are multiple arcs outside the sector, then all items in this sector are interested by all persons represented by these arcs. In this way, we can show a person's interested items, and the set relations between people according to their interested items. If the arcs outside the ring region do not overlap, these people have no shared interests. According to the guideline given in [30] for using color for labeling, less than a dozen colors should be used, and thus our radial display cannot display many user groups simultaneously. In this paper, we refer to items residing in the ring region as "anchored" items, since their positions are fixed when performing the layout of the items in the focus region.

Focus Region The region inside the inner circle is called the focus region and can be used for several purposes. At beginning, all items are put in the inner circle using MDS. After users select some people in the social network and their interested items are positioned in the ring region, our system will recommend some similar or relevant items and put them inside the inner circle.

Moreover, instead of using arcs to represent set memberships, the interested items of a person or a group of people can be represented by directly coloring the items in the focus region and the ring region. By observing the distributions of the colored items relative to the arcs, the aforementioned patterns of “relatedness” can be detected.

The anchored items will be positioned in the ring region according to similarity or relevance and similar items will be placed near one another. This feature allows users to find item clusters within each group and explore the suggested items related to different clusters.

The size and transparency of the nodes can be used to encode additional information that is deemed as useful in the specific application scenarios, or to encode a degree of interests.

If we compare Fig. 5 and Fig. 6, we can clearly see the difference between our method with the Venn Diagram and BubbleSet. Our design offers several advantages: 1) Radial layout is widely adopted and there exist effective methods to layout items according to their distance to the items on the ring. Thus, we can easily position suggested items in the inner circle and it is intuitive for users to understand the encoding scheme. 2) Adjustable inner circle offers great flexibility for users. A larger inner circle will focus more on the suggested items while a smaller inner circle will leave more space for highlighted items. 3) Using the spatial relationships between line segments to show the set relations is also intuitive.

4.2.3 User Interactions

Rich user interactions are provided in our system to support more flexible exploration of the data.

Zooming The size of the inner circle can be adjusted to give different ratio for the focus region and the ring region, which simulates a zoom-in/zoom-out effect for the focus region (Fig. 6).

Arc swapping Our method allows best comparisons for the most inside 2 or 3 layers of arcs. Users can always swap the arcs to bring their most interested 2 or 3 groups inside for better comparisons (Fig. 6).

Filtering To reduce the visual clutter, nodes placed in the focus region can be filtered based on the degree of interest (DOI) [10]. A DOI function [29] could be used. This is composed of two parts: *a priori* importance function based on some intrinsic properties of the nodes, and a distance function which is the shortest distance from the nodes to the set of anchored items. The threshold can be adjusted such that an appropriate number of nodes could be displayed within the circle and the visual clutter could be reduced.

Animated transition When the user performs arc swapping or zooming which needs reconfiguration of the view, we apply animated transitions. For example, in arc swapping, when a layer of arcs is brought inside by the user, other layers will be pushed outward, and visually it will look like “ripple”.

5 Implementation

In this section, we will briefly describe the layout algorithm used for the radial view. In the radial view, the layout takes two steps: 1) arrange the “anchors” (i.e., items interested by selected groups) in the ring region, and 2) place the “free nodes” (e.g., the suggested items) within the inner circle.

We take into consideration the following aesthetic criteria when computing the layout of the items:

a) Anchored items similar to each other are placed at nearby angular positions in the ring.

b) The number of arc segments denoting the set memberships on the first layer should be as small as possible to reduce visual clutter.

c) The "free nodes" are positioned close to the anchored items which are similar to them.

In the following description, S_a denotes anchored items; S_f denotes free nodes; and S_a and S_f are disjointed subsets of S.

Arrange the anchored items An initial layout of items in S_a is obtained through the mean and median iterations step of the circular layout algorithm proposed in [11]. We consider the similarity measure between the items by weighting the edge lengths in the optimization function.

After the initial layout step, the spatial closeness of the items should reflect their similarity, and the items are distributed at irregular intervals in the ring region since the mean- and median iteration step gives a continuous approximation of the node positions [11]. Therefore we compute an angular ordering of the items from the initial layout and redistribute the nodes at regular intervals. Assume that after this step, we obtain an ordering of the items either clockwise or counterclockwise which can be expressed as a function $\pi(s_i) : s_i \to 0, 1, \ldots, n-1$ where $n = |S_a|$. Directly using this ordering to place the items may result in a lot of arc segments even in the first layer, which is undesirable due to the visual clutter it causes. Therefore in the second step, we use a heuristic which partially retains the original ordering meanwhile reducing the number of arc segments on the first layer. The angular distance (gap) between two items in the ring region can be approximated as the minimum one of their clockwise distance and counterclockwise distance:

$$d(s_i, s_j) = \min((\pi(s_i) - \pi(s_j) + n) \bmod n, (\pi(s_j) - \pi(s_i) + n) \bmod n)$$

We define the following energy function:

$$E(\pi) = \sum_{s_i, s_j \in S_a} \omega_{ij} d(s_i, s_j) + \lambda f(k(\pi)) \quad (1)$$

This energy function indicates the trade-off between the first two aesthetic criteria, where ω_{ij} is the similarity between s_i and s_j, λ is the weighting coefficient, $k(\pi)$ denotes the number of arc segments on the first layer given π as the angular ordering of the items, and f could be any monotonously increasing function. We use $f = k^2$ in our experiments.

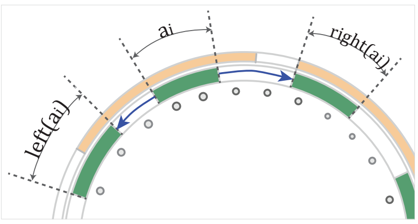


Fig. 7 After an initial layout, the anchored items are arranged according to their similarity to each other, however, this may result in a lot of arc segments even on the first layer (the green arcs in the figure). Therefore we can merge some arc a_i with its left or right neighbor.

Starting from the initial layout, we apply a greedy heuristic which merges two nearby arc segments and rearranges the anchored items accordingly at each step to minimize the energy function. The pseudo code is given as below, where $A = \{a_0, a_1, a_2, \ldots, a_m\}$ is the set of arcs on the first layer resulted from the ordering π. Each a_i has a left neighbor $left(a_i) \in A$ and a right neighbor $right(a_i) \in A$ (Fig. 7). Each a_i can either be merged to $left(a_i)$ or $right(a_i)$, and we denote $dE(\pi_{a_i})$ as the larger descent of E by merging a_i either to the left or right. Note that $dE(\pi_{a_i})$ can be minus, if $E(\pi_{a_i})$ is less than current $E(\pi)$.

Algorithm 1 MergeArcs(S_a, π, A)

while $|A| \geq 2$ **do**
  $i \leftarrow$ value i of $min(dE(\pi_{a_i}))$
  if $dE(\pi_{a_i}) < 0$ **then**
    merge a_i with $left(a_i)$(or $right(a_i)$)
    update π
    A.remove(a_i)
    compute $dE(\pi_{a_j})$ for $\forall a_j \in A$
  else
    return π
  end if
end while
return π

Embedding free nodes The aesthetic criterion of embedding the nodes within the circle is that their spatial closeness to the radial anchors should reflect their similarity to anchored nodes, as in [20]. We employ Barycenter method [2] with a modified position function illustrated in Equation 2. The position p_{s_i} of free node s_i iteratively moves until it becomes the geometric center of all the around nodes s_j and s_k. An extra weight γ can be added to draw the free nodes close to the anchored items similar to them with stronger "forces".

$$p_{s_i \in S} = \frac{\sum_{s_j \in S_f} \omega_{ij} p_{s_j} + \sum_{s_k \in S_a} \gamma \omega_{ik} p_{s_k}}{\sum_{s_j \in S_f} \omega_{ij} + \sum_{s_k \in S_a} \gamma \omega_{ik}} \tag{2}$$

6 Experimental results

In this section, we demonstrate the effectiveness of our visualization framework through the experiment on a real dataset. The dataset is retrieved from Last.fm[1], a social music service website which maintains a catalog of artists, albums, and tracks. Users of the website can listen to music, setup personal profiles of artists they like, and add other users as friends. Last.fm also provides a webservice API, which can be used for querying about data of the users and the artists, such as the tags of each artist, the total number of listeners of each artist, and the similarity between two artists.

In this experiment, our system is used for interactive visual recommendation and discovery of music artists based on this Last.fm dataset released in [3] and the artist similarity data obtained through the Last.fm webservice API. The Last.fm dataset contains the friendship graph of the users (the names of the users have been anonymized) and user listening information which is a bipartite relation between the users and the artists. We selected the most popular 500 artists from the dataset (which originally contains 17632 artists) according to the total listening count of each artist and obtained the similarity measures between the music artists through the webservice API. For each artist, we retrieved a list of 25 most similar

[1] http://www.last.fm/

artists to her and the corresponding similarity measure, a score from 0.0 to 1.0 as provided by Last.fm. The adjacency lists are used to build a weighted similarity graph of the artists. The inverse of the similarity measure are used as the distance when computing DOI of the artists. From the user friendship graph, we substracted a BFS tree rooted at any specified user node. Thus we can emulate the process as a user of the website visits her neighbors through hyperlinks. For each user, we obtained a list of artists that she likes by thresholding on the listening counts.

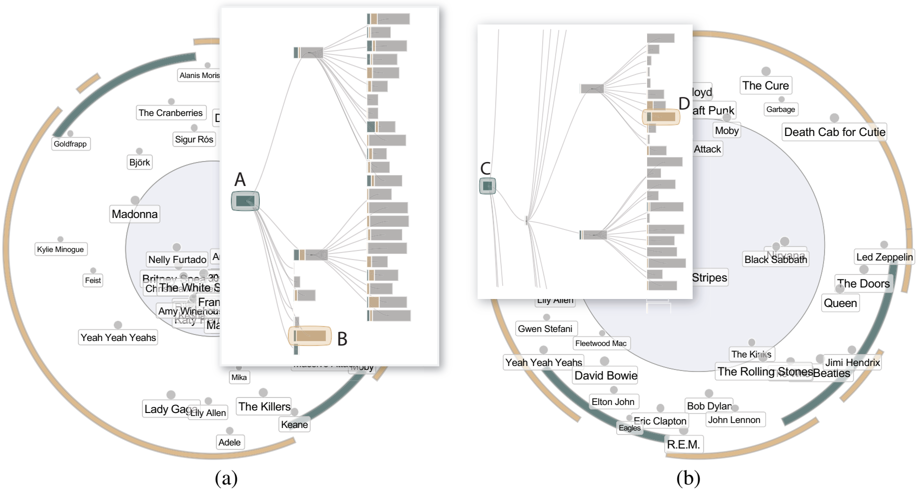


Fig. 8 (a) Two people who are friends but have quite different interests; (b) Two people which are not friends but have similar interests.

Our system can help users explore many kinds of information in bipartite relationships between a social network and a set of items. A typical and natural usage of our visual recommendation system starts from selecting one's own node (the root node). After the selection, the bar chart and the radial view will be updated accordingly, showing the number of shared interests of the neighborhoods with the root node. Fig. 8(a) shows that after the selection, the user find that some of her immediate neighbors do not share much interest with her, because all the neighbor nodes of node A have a small size of peacock blue bar in their barcharts. She might be curious about those artists and may want to try herself. Therefore she selects a node representing her close friend and updates the radial view again to see who are the artists that her friend likes. The user may also find that some distant users share more interest with her than the immediate neighbors (Fig. 8(b)), because the length of the peacock blue bar in node D is the largest one among all the other nodes including immediate neighbor nodes. She may want to know who are the artists that they both like. Therefore she also selects on the corresponding node and find that they both like artists include "Jimi Hendrix", "The Doors", "David Bowie" (artists active during approximately 60s and 70s, and are mostly tagged as "classic rock" on the website).

In the radial view, the nodes representing the artists on the ring are placed such their angular distances reflects their similarities. This can reveal some clusters of similar artists. From Fig. 9(a), we can see that there are two groups of artists included in the user's listening profile, a group of female vocalists tagged with "pop", "dance", etc., and a group of music

(a) (b) (c) (d)

Fig. 9 (a) Find similar artists to the users tastes, which might include multiple clusters; (b) Adjust the size of the inner circle to find more suggested artists or to get a more clear overview of the artists in one's own profile; (c) Find users who are interested in some items in the social network; (d) Highlight the interests of a third person in the radial view. The user can observe the relative spatial distribution.

artists tagged with "rock", "pop punk", etc. The user can therefore select the suggested items based on which cluster the item is more close to, as marked by orange. Our system provides users with the flexibility of adjusting the size of the inner circle to look at more suggested artists or to get a more clear overview of her own profile. The left one in Fig. 9(b) illustrates that when the inner circle is contracted, the user can get a better overview of his interests, but the suggested items are cluttered. Therefore he can enlarge the inner circle and the suggested items will be more clearly showed to the users (right one in Fig. 9(b)). He may select several music artists like "50 Cent" "T.I" (tagged as "rap" and "hip-hop"), which are very similar to "Eminem", or music artists like "Adele", "Destiny's Child", which are both female vocalists.

As the user select some suggested artists on the radial view, the other users who are interested in the artists will be highlighted in the tree view. In Fig. 9(c), the user selects "Mariah Carey", and find that a lot of his neighbors are interested in the artist (lots of nodes have visible peacock blue bars), so perhaps she can give a try.

In Fig. 9(d), the interests of a third person are highlighted with coral pink color in the focus and the ring region. Some interesting patterns can be observed from the figure. The person has more common interests with the person denoted by the peacock blue arcs than with the person denoted by the chrome yellow arcs, because most coral pink nodes fall into the fan shaped areas of the peacock blue arcs. They both like music artists like "Aphex Twin", and "Deftones", etc. It could also be found that the two people in particular share two groups of interests: one group tagged by "ambient", "electronic", "chillout", and another group tagged by "alternative rock", "metal", "electronic", etc. From this experiment, we can see that our visualization framework is an effective technique to simultaneously reveal all kinds of bipartite relationships including the social relation between people, the set relation between people's interested items, the similarity or relevant relations between items in the set, and the association relation between the people in the social network and the items in the set.

Our system is implemented in Java with the Prefuse[2] information visualization toolkit. The system runs on a laptop with Intel Core 2 Duo CPU and 4GB memory.

7 Conclusions

In this paper, we have presented a visualization system to help explore the bipartite relations between a graph and a set. Our system adopted a linked view design. We proposed two novel visual encoding schemes: an enhanced graph view and a radial view. The enhanced graph view illustrates a social network of people and statistical information about the interested items of people. The radial view can help explore the items interested by several users or groups, and the corresponding set relations. Also it displays related similar items which can be applied for recommendation. Our system supports bi-directional bipartite relation explorations: users can start from the social network to find the related items in the set; or they can start from items in the set to find people in the social network who are interested in them.

There are several avenues for future work. Our system faces the scalability issue for large social network and interests set. The radial layout can only show the relations of 3 sets. We plan to provide some overview or statistics to guide the exploration. We also want to integrate community detection into our system.

References

1. B. Alper, N. H. Riche, G. Ramos, and M. Czerwinski. Design study of LineSets, a novel set visualization technique. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2259–67, Dec. 2011.
2. G. Battista. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.
3. I. Cantador, P. Brusilovsky, and T. Kuflik. 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In *Proceedings of the 5th ACM conference on Recommender systems*, RecSys 2011, New York, NY, USA, 2011. ACM.
4. N. Cao, D. Gotz, J. Sun, Y.-R. Lin, and H. Qu. Solarmap: Multifaceted visual analytics for topic exploration. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 101 –110, dec. 2011.
5. N. Cao, J. Sun, Y.-R. Lin, D. Gotz, S. Liu, and H. Qu. Facetatlas: Multifaceted visualization for rich text corpora. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1172 –1181, nov.-dec. 2010.
6. C. Collins, G. Penn, and M. S. T. Carpendale. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1009–1016, 2009.
7. W. Cui, H. Qu, H. Zhou, W. Zhang, and S. Skiena. Watch the story unfold with textwheel: Visualization of large-scale news streams. *ACM Trans. Intell. Syst. Technol.*, 3(2):20:1–20:17, Feb. 2012.

[2] http://prefuse.org/

8. M. Dörk, S. Carpendale, and C. Williamson. EdgeMaps: visualizing explicit and implicit relations. In *VDA2011: Conference on Visualization and Data Analysis*, pages 1–12, 2011.
9. W. Freiler, K. Matkovic, and H. Hauser. Interactive visual analysis of set-typed data. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1340–1347, 2008.
10. G. W. Furnas. Generalized fisheye views. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '86, pages 16–23, New York, NY, USA, 1986. ACM.
11. E. R. Gansner and Y. Koren. Improved circular layouts. In *Proceedings of the 14th international conference on Graph Drawing*, GD'06, pages 386–398, Berlin, Heidelberg, 2007. Springer-Verlag.
12. M. Ghoniem, J.-D. Fekete, and P. Castagliola. A comparison of the readability of graphs using node-link and matrix-based representations. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 17–24, 2004.
13. J. Heer and A. Perer. Orion: A system for modeling, transformation and visualization of multidimensional heterogeneous networks. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, 2011.
14. N. Henry and J.-D. Fekete. Matrixexplorer: a dual-representation system to explore social networks. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):677–684, 2006.
15. N. Henry, J.-D. Fekete, and M. J. McGuffin. Nodetrix: a hybrid visualization of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, 2007.
16. E. Kandogan. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, pages 107–116, New York, NY, USA, 2001. ACM.
17. H. Kang, C. Plaisant, B. Lee, and B. B. Bederson. Netlens: iterative exploration of content-actor network data. *Information Visualization*, 6:18–31, March 2007.
18. B. Lee, M. Czerwinski, G. Robertson, and B. B. Bederson. Understanding research trends in conferences using paperlens. In *CHI '05 extended abstracts on Human factors in computing systems*, CHI EA '05, pages 1969–1972, New York, NY, USA, 2005. ACM.
19. Z. Liu, S. Navathe, and J. Stasko. Network-based visual analysis of tabular data. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 41 –50, oct. 2011.
20. K. Misue. Drawing bipartite graphs as anchored maps. In *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation - Volume 60*, APVis '06, pages 169–177. Australian Computer Society, Inc., 2006.
21. K. Misue and Q. Zhou. Drawing semi-bipartite graphs in anchor+matrix style. In *Proceedings of the 2011 15th International Conference on Information Visualisation*, IV '11, pages 26–31, Washington, DC, USA, 2011. IEEE Computer Society.
22. A. Naud, S. Usui, N. Ueda, and T. Taniguchi. Visualization of Documents and Concepts in Neuroinformatics with the 3D-SE Viewer. *Frontiers in neuroinformatics*, 1:7, Jan. 2007.
23. H. Schulz, M. John, A. Unger, and H. Schumann. Visual analysis of bipartite biological networks. In *Eurographics Workshop on Visual Computing for Biomedicine*, 2008.
24. J. Sharko, G. Grinstein, and K. Marx. Vectorized radviz and its application to multiple cluster datasets. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1444 –1427, nov.-dec. 2008.
25. Z. Shen, K.-L. Ma, and T. Eliassi-Rad. Visual Analysis of Large Heterogeneous Social Networks by Semantic and Structural Abstraction. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1427–39, 2006.
26. P. Simonetto, D. Auber, and D. Archambault. Fully automatic visualisation of overlapping sets. *Computer Graphics Forum*, 28(3):967–974, 2009.
27. J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, Jan. 2008.
28. K. Thiel, F. Dill, T. Kotter, and M. Berthold. Towards visual exploration of topic shifts. In *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*, pages 522 –527, oct. 2007.
29. F. van Ham and A. Perer. "search, show context, expand on demand": Supporting large graph exploration with degree-of-interest. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):953–960, Nov. 2009.
30. C. Ware. *Information Visualization: Perception for Design (Interactive Technologies)*. Morgan Kaufmann, 1st edition, Feb. 2000.
31. M. Wattenberg. Visual exploration of multivariate graphs. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 811–819, New York, NY, USA, 2006. ACM.