Singapore Management University
Institutional Knowledge at Singapore Management University

Research Collection School Of Information Systems   School of Information Systems

ViStruclizer: A Structural Visualizer for Multi-dimensional Social Networks

Bingtian DAI
Singapore Management University, btdai@smu.edu.sg

Agus Trisnajaya Kwee
Singapore Management University, aguskwee@smu.edu.sg

Ee Peng LIM
Singapore Management University, eplim@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Databases and Information Systems Commons, and the Numerical Analysis and Scientific Computing Commons

Citation

ViStruclizer: A Structural Visualizer for Multi-dimensional Social Networks

Bing Tian Dai, Agus Trisnajaya Kwee, and Ee-Peng Lim

Living Analytics Research Centre
Singapore Management University
80 Stamford Road, Singapore 178902
{btdai,aguskwee,eplim}@smu.edu.sg

Abstract. With the popularity of Web 2.0 sites, social networks today increasingly involve different kinds of relationships among different types of users in a single network. Such social networks are said to be multi-dimensional. Analyzing multi-dimensional networks is a challenging research task that requires intelligent visualization techniques. In this paper, we therefore propose a visual analytics tool called *ViStruclizer* to analyze structures embedded in a multi-dimensional social network. *ViStruclizer* incorporates structure analyzers that summarize social networks into both node clusters each representing a set of users, and edge clusters representing relationships between users in the node clusters. *ViStruclizer* supports user interactions to examine specific clusters of users and inter-cluster relationships, as well as to refine the learnt structural summary.

1 Introduction

Web 2.0 sites widely adopt online social networks as the means to connect their users with one another to encourage users to participate in information sharing (e.g., friendship network in Facebook[1]) as well as to collaborate with one another (e.g., collaboration networks in Wikipedia[2]). Unlike traditional social networks which involve a single type of nodes or actors and a single relation type, these online social networks are *heterogeneous* and of *large scale*, where multiple types of nodes and relations may exist in the same network.

In network analysis, the term "mode" refers to a distinct set of entities [14], a network with different types of objects is therefore called a **Multi-Mode Network**.

Example 1. An online social network provides a platform for friends to share photos and videos. There are three types of objects in this network, namely people, photos and videos. This network is therefore a multi-mode network.

We would like to make a remark about the modes here. The modes of nodes give an explicit grouping of the nodes. This grouping is however, not always

---

[1] www.facebook.com

[2] www.wikipedia.org

informative as there are other implicit groupings of nodes that are more interesting. The implicit structure to be discovered should be orthogonal to the modes of the nodes, and it reveals the implicit groups from the observed interactions among the nodes. Hence, for multi-mode networks, it is important to extract their implicit structures even though the modes are known.

There are also networks with different types of relations, each representing one type of interaction. It is often that multiple relations co-exist between a pair of individuals. For example, a user in the network can send a message to another user, or comment on another user's status. Since each type of interaction suggests a different association between two users [5], if we had only one relation to represent different types of interactions, there would have been too much loss in the network information. Therefore, we define a **Multi-Relational Network** to be one that describes the relationship from one object to another as a composition of different relations.

Example 2. Individuals in the social network, mentioned in Example 1, send messages or comments to one another. Each user may also upload a photo or a video, or comment on others' photos or videos. There may exist some interaction links between a photo and a video, if they are taken at the same location, or the photo is captured from the video. This network is illustrated by Figure 1, where multiple types of relations and objects can be found.
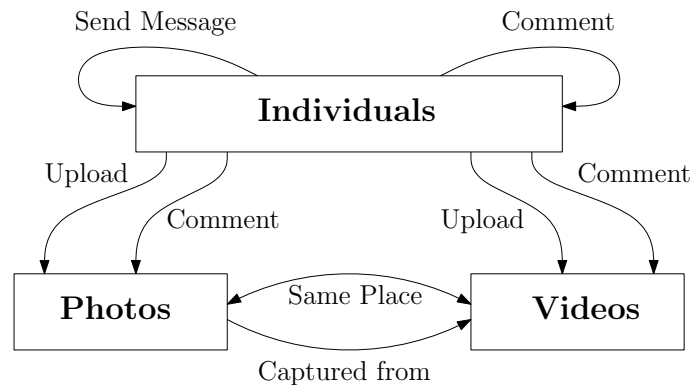
Fig. 1. A Multi-Dimensional Social Network

The network in Figure 1 is multi-mode and multi-relational at the same time. A network with heterogeneous types of nodes and relations is therefore regarded as a **Multi-Dimensional Network**. Multi-mode networks and multi-relational networks are just special cases of multi-dimensional networks. The analysis of multi-dimensional networks is known to be harder than simple networks and is currently an active research topic [4].

As social networks grow in size, they become more difficult to analyze as there are many nodes and relation edges. A typical user can only handle less than a hundred of nodes and edges. Beyond that, visualization techniques have to be used [2,8,11]. These techniques usually summarize large social networks into smaller and simpler networks that are human-interpretable. This summarization process essentially groups nodes and edges in an original network into a simple network of node clusters and relationships between node clusters. The grouping however is not arbitrary. It ought to preserve information contained in the original network. Otherwise in the extreme case, the summarized network may consist of only one node cluster containing all nodes and such a summarized network does not help user-interpretation.

Many existing network visualization and summarization techniques [10,7,13,16] are designed mainly for simple social networks, but not for multi-dimensional networks. Recent works, OntoVis[12] and FacetAtlas[3], focus on visualizations of heterogeneous networks. However, OntoVis is mainly designed for multi-mode networks, as edge types are determined by node types and multiple types of relations between two nodes are not taken into consideration. FacetAtlas visualizes multi-dimensional networks in multiple facets. But again, their nodes are connected by at most one type of relation, i.e., an edge within a facet.

In this paper, we focus on visualizing multi-dimensional networks by designing and implementing a network visual analytics system, called ***ViStruclizer***, which presents a summarized network structure from a multi-dimensional network as an overlay over the original set of nodes and relations for human users to interpret. Given that multi-mode and multi-relational networks are special classes of multi-dimensional networks, *ViStruclizer* also works well on these networks.

In the absence of research in visualizing multi-dimensional networks, we define two key research objectives for the *ViStruclizer* project. The first objective is to introduce a design framework that can be adopted by *ViStruclizer* and other future visual analytics systems for multi-dimensional networks. The framework identifies the required system components, their corresponding functionalities and how these components interact with one another. The second research objective is to develop the working visual analytics system and to demonstrate features that support the important visualization requirements. To meet the above objectives, we have three major contributions in this paper, namely:

- We propose a network visualizer system design framework that covers the essential system modules and the desired functions for analyzing a multi-dimensional network. The design framework also defines the network structure representation that summarizes the original network.
- We have developed a visualizer called *ViStruclizer* to realize our proposed system design framework. *ViStruclizer* is capable of deriving a network structure from a large multi-dimensional social network, and allowing users to examine and navigate the derived network structure. The structure captures the positions of the individuals and their roles in the network.

– We propose a means in *ViStruclizer* to allow users to exercise their own judgments on the structures of social networks and to refine the automatically learnt network structures according to the users' preferences. For example, one may want to correct an individual's position in the learnt network structure. This is achieved by making *ViStruclizer* a *semi-supervised* system.

2   The *ViStruclizer* System Design Framework

In this section, we discuss the system design framework of *ViStruclizer* for multi-dimensional networks in three parts. We first describe the representation of multi-dimensional networks in some high level summary networks in Section 2.1. This is followed by the architecture framework of *ViStruclizer* and its components (see Section 2.2). We finally outline the interactive visualization requirements to be supported by *ViStruclizer* in Section 2.3.

2.1   Summary of Multi-Dimensional Networks

Let $G = (V, E, M, R, m, r)$ denote a multi-dimensional network, where V is the set of vertices, E is the set of edges, $M = \{M_1, \ldots, M_t, \ldots\}$ is the set of vertex modes and $R = \{R_1, \ldots, R_s, \ldots\}$ is the set of relations. The second last element of G, $m : V \to M$, is the mapping from vertices to their modes; each vertex V_i has a definite mode $m(V_i)$. Compared to the vertex modes, it is more complicated to represent the edges since an edge is a combination of different type of relations. Thus the last element of G, $r : E \to R_1 \times \cdots \times R_s \times \cdots$, is defined as a mapping from edges to the Cartesian product on R, i.e., $r(E_{i,j}) = (r_1, \ldots, r_s, \ldots)$, each r_s is the number of times the relation R_s is observed from vertex V_i to V_j. For example, $r(E_{i,j}) = (1, 1, 2)$ indicates that $E_{i,j}$ consists of one occurrence of relation R_1, one occurrence of R_2, and two occurrences of R_3. We also denote the value of $r(E_{i,j})$ on relation R_s by $r_s(E_{i,j})$.

When the network has large number of vertices and edges, summarizing them by a representative high level network structure becomes necessary. For example, one way to summarize a social network is to group individuals with similar connections into one community, and many social network visualizers focus on density-based community structures, e.g., Vizter [7] and HiMap [13].

As edges are heterogeneous in multi-dimensional networks, not only is it necessary to group similar vertices, but similar edges are also expected to be grouped. *Positional and Role Analysis* groups individuals into one position if they behave similarly, and similar relationships into roles. The structure derived by this analysis treats relationships across different communities as important as relationships within a community, which is more informative than community structures. White, Boorman and Breiger [15] observed that the most informative positional and role analysis requires multiple ties, i.e., a multi-relational network, and they introduced the concept of *blockmodel* to study roles and positions. Wasserman and Faust [14] defined a blockmodel as a partition of vertices into k positions (vertex clusters). Roles from one position to another are modeled

by blocks (edge clusters). Airoldi [1] extended blockmodels to mixed membership blockmodels (MMB), where the position of each vertex is modeled by a probabilistic distribution. Dai, Chua and Lim [5] generalized MMB [1] further on multi-relational networks. The blockmodel developed in [5] is called *Generalized Stochastic Blockmodels* (GSBM), which models (i) the position of each vertex by a probabilistic distribution; and (ii) each block by a multivariate distribution over the set of relations.

Without loss of generality, we assume a structural summary for a multi-dimensional network provides two kinds of clusters, vertex clusters and edges clusters. $C = \{C_1, C_2, \ldots, C_k\}$ is the set of vertex clusters; for vertex $V_i \in V$, $\pi_{i,u}$ is the membership probability of V_i assigned to cluster C_u, subject to $\sum_{u=1}^{k} \pi_{i,u} = 1$. The *weight of the cluster* C_u is the accumulated probability of all vertices assigned to C_u, i.e.,

$$w(C_u) = \sum_{i=1}^{|V|} \pi_{i,u} \tag{1}$$

The *weight of mode* M_t *of the cluster* C_u is proportional to the accumulated probability of all vertices of mode M_t assigned to C_u, i.e.,

$$w(M_t|C_u) \propto \sum_{m(V_i)=M_t} \pi_{i,u} \tag{2}$$

We also define $B = \{B_{u,v}|C_u, C_v \in C\}$ as the edge clusters from C_u to C_v in the structural. Similarly, $\rho_{i,j,u,v}$ is the membership probability of edge $E_{i,j}$ assigned to $B_{u,v}$, subject to $\sum_{u,v=1}^{k} \rho_{i,j,u,v} = 1$. The *weight of an edge cluster* $B_{u,v}$ is the accumulated probability of all its edges, that is:

$$w(B_{u,v}) = \sum_{E_{i,j}} \rho_{i,j,u,v} \|r(E_{i,j})\| \tag{3}$$

where $\|r(E_{i,j})\| = \sum_{R_s \in R} r_s(E_{i,j})$ is the *1-norm* of $r(E_{i,j})$ since all $r_s(E_{i,j})$ are positive. $\|r(E_{i,j})\|$ can also be other form of norm. The *width* of an edge cluster $B_{u,v}$ is logarithmically proportional to the weight of the edge cluster. The weight of relation R_s of the edge cluster $B_{u,v}$ is proportional to the accumulated probability of relation R_s, i.e.,

$$w(R_s|B_{u,v}) \propto \sum_{E_{i,j}} \rho_{i,j,u,v} r_s(E_{i,j})$$

This is valid with $\|r(E_{i,j})\|$ being the 1-norm, and it ought to be altered for other kinds of norm.

2.2   Architecture Framework

ViStruclizer takes a multi-dimensional social network as input, and its primary objective is to visualize the network through its structural summary. With vertices and edges grouped into clusters with some latent semantics, it helps users to understand the original network.

The architectural framework of *ViStruclizer* is shown in Figure 2. There are three components in *ViStruclizer*, namely the structure analyzer, the back-end database and the visualizer which serves as the frontend interface of *ViStruclizer*.

Our objective is to make *ViStruclizer* general enough to accommodate different structure analyzers that can return different structural summaries. The structure analyzer in *ViStruclizer* either takes input directly from a multi-dimensional network or reads the network from the database, and constructs a structural summary to be presented by the visualizer. The structural summary consists of a mandatory element and another optional element, which will be discussed shortly.

The visualizer component in *ViStruclizer* takes both the original multi-dimensional network from the database and the structural summary as input and presents the network to the users, as shown by the solid arrows in Figure 2. The dash-dotted arrows show that the users can provide their feedback to *ViStruclizer*. The user feedback allows the structure analyzer to further refine the structural summary, and presents the network using refined structural summary to the users.
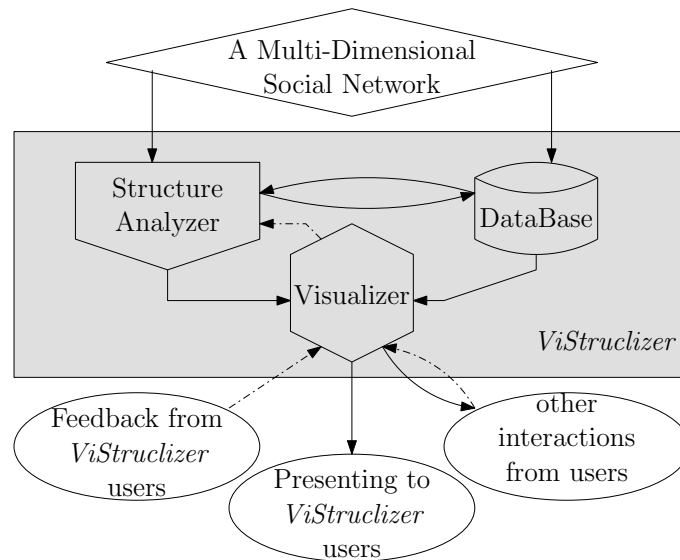

Fig. 2. The System Design of *ViStruclizer*

Within the structural summary returned by the structure analyzer, the mandatory element refers to the cluster membership for all vertices in the original network. If a structure analyzer returns a hard clustering, the cluster membership for a vertex is then its cluster indicators, i.e., cluster id. In *ViStruclizer*, we consider mixed cluster memberships derived by soft clustering or fuzzy clustering techniques, e.g. EM algorithm [9], which are widely adopted in cluster analysis. The mixed cluster memberships are typically represented in the form of probabilistic distributions, i.e., a vertex is assigned to multiple clusters with different probabilities. In Section 2.1, such probabilities are denoted by $\pi_{i,u}$, which is the

probability of assigning vertex V_i and cluster C_u, subject to $\sum_{u=1}^{k} \pi_{i,u} = 1$. Hence the cluster memberships in the structural summary are probabilistic distributions for vertices assigned to vertex clusters.

The optional element for the structure analyzer is to return the cluster memberships for the edges. Similar to the vertex cluster membership, the cluster membership of an edge indicates between which pair of clusters the edge belongs to. If k vertex clusters are extracted by structure analyzer, there exist k^2 edge clusters, representing every possible pairs of k vertex clusters.

If a hard clustering for the vertices is adopted, for an given edge, its edge cluster is defined by the cluster of the source vertex to the cluster of the target vertex. However, when vertices are associated with vertex clusters with mixed memberships, the membership distributions of the edges belonging to an edge cluster become more complicated. This is denoted by $\rho_{i,j,u,v}$ in Section 2.1, which is the cluster membership of edge $E_{i,j}$ assigned to cluster pair (C_u, C_v), subject to $\sum_{u,v=1}^{k} \rho_{i,j,u,v} = 1$. This issue will be discussed in greater detail in Section 3. For now, we would like to design *ViStruclizer* to be able to cope with both kinds of edge memberships.

2.3   Interactive Visualization Requirements of *ViStruclizer*

For *ViStruclizer* to be an effective tool for analyzing multi-dimensional networks interactively, it has to satisfy a few essential visualization requirements mentioned below:

Network Search. Finding a user vertex in a large multi-dimensional network is like searching a needle in the haystack. Vertex search by label and other attributes is therefore essential. Once some target vertices are found, they can serve as the lead to explore the rest of network. When structural summary is available, vertex search can also return the vertex clusters of vertices meeting the search criteria and help users to determine the relevance of these vertices.

Summary Network Visualization. A summary network consists of vertex clusters and edge clusters, and each cluster has its own salient attributes, i.e., cluster weights, cluster modal weights. These clusters and their attributes should be visualized clearly using graphical properties such as shape, color and size. The composition of vertex types and edge types in clusters is also another piece of information to be visually presented clearly. In the visualizer component in *ViStruclizer*, the size of a vertex cluster is determined by its weight. The width of an edge cluster is also logarithmically proportional to the weight of the edge cluster. We use multiple colors to distinguish different kinds of vertices and relations in a multi-dimensional network. An edge cluster is thus represented by a directed multi-color edge from the source vertex cluster to the target vertex cluster such that each color represents a different relation, and the proportion of each color represents the weight of the corresponding relation.

Summary Network Exploration. For a user to find out how vertices are grouped into vertex clusters, *ViStruclizer* has to support user interactively expanding or collapsing vertex clusters. Such a visual operation can be non-trivial for very large vertex clusters each with too many vertices for the user to examine. The main challenge of summary network exploration is therefore to expand the cluster without being overwhelmed by the large number of vertices. This can be achieved by selectively expanding a vertex cluster as opposed to complete vertex cluster expansion. When vertex clusters are derived by soft clustering, one can use a probabilistic threshold to control the extent of vertex cluster expansion by expanding only the vertices with membership probabilities above this threshold. This threshold can be set globally for all vertex clusters, or locally for only one vertex cluster.

Structure Refinement. Structural summary automatically learnt from multi-dimensional networks may not always match user expectation. When a user disagrees with the way a structural summary summarizes the underlying network, she may want to refine the cluster membership distribution of a vertex. Such a refinement will require the structural summary to be revised based on user input. We therefore require the structure analyzer to be *semi-supervised*. Depending on how the structure analyzer clusters the vertices, the change in one vertex's membership distribution may cause change in others' membership distributions. As the structure analyzer refines the summary structure, the visualizer has to update the summary network accordingly with three kinds of interactions: i) change the membership distribution of one or more vertices; ii) create a new cluster, and indicate several members of the cluster; and iii) merge two existing clusters.

3   Case Study for Structure Analysis and Visualization

Based on our proposed framework, a working *ViStruclizer* has been developed with its visualizer and structure analyzer components implemented in Javascript and C++ respectively. *ViStruclizer* uses MySQL database system for backend storage. In this section, we demonstrate the capabilities of *ViStruclizer* using a multi-dimensional network extracted from IMDb.

3.1   The IMDb Network and Its Structure

We focus on the people involved in the movie industry and their network. There are more than 4 million of them and their network is very sparse. In our case study, we selected a denser subset of the network as follows. We started with a set of eleven directors, James Cameron, Chris Columbus, Jon Favreau, Ron Howard, Doug Liman, Christopher Nolan, Guy Ritchie, Martin Scorsese, Steven Soderbergh, Steven Spielberg and David Yates. These directors directed 73 movies from year 2000 to year 2010. We then further expanded to 486 people, including directors, producers and actors/actresses, who are involved in at least
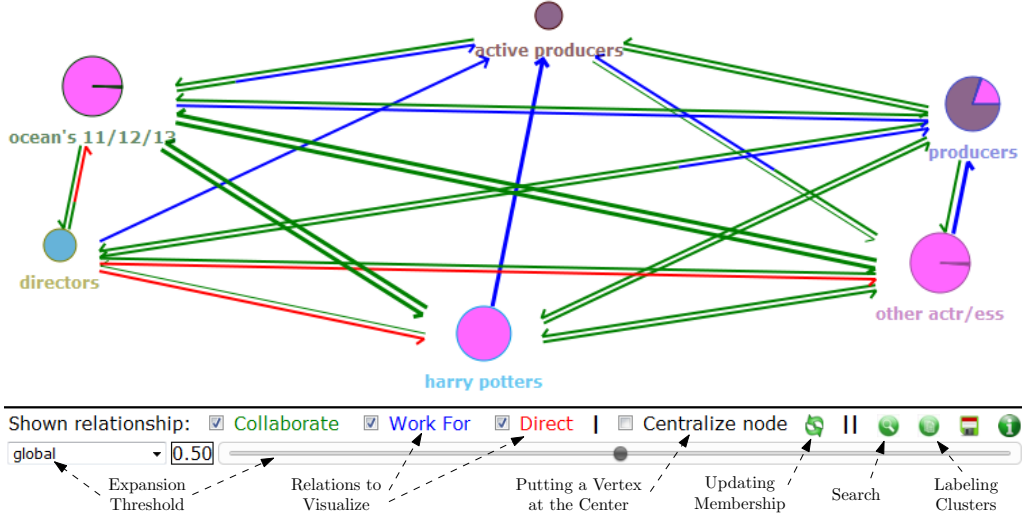
Fig. 3. The Structure of IMDb Network by *ViStruclizer*

two of the 73 movies. There are 3 modes for people in this network, namely, `actors/actresses`, `directors` and `producers`.

We also established three relations among the 486 people, i.e., `collaborate`, `direct` and `work_for`. For each movie, relation `collaborate` is observed among actors/actress, among directors, or among producers; relation `direct` is from directors to their directed actors/actresses; and relation `work_for` is observed from both actors/actresses and directors to producers. Therefore, this IMDb network is a multi-dimensional network with three modes and three relations.

We then incorporate the structure analyzer GSBM into *ViStruclizer* to visualize the extracted IMDb network. Empirically, we set the number of vertex clusters in the resultant structural summary to be $k = 6$. Besides grouping vertices into six vertex clusters, GSBM also returns edge clusters membership distributions, which is the optional element mentioned in Section 2. In GSBM, the probability of observing $r(E_{i,j})$ in edge cluster $B_{u,v}$ is given by

$$p(r(E_{i,j})|B_{u,v}) = \prod_{s=1}^{h} \text{Pois}(r_s(E_{i,j})|B_{u,v,s})$$

where $B_{u,v,s}$ is the Poisson parameter of edge cluster $B_{u,v}$ on relation R_s. Therefore, the probability of $r(E_{i,j})$ being modeled by $B_{u,v}$ is

$$\rho_{i,j,u,v} = p(u,v|r(E_{i,j}), B) \propto p(u,v,r(E_{i,j})|B) = \pi_{i,u}\pi_{j,v}p(r(E_{i,j})|B_{u,v})$$

Hence, the edge cluster membership distribution for edge $E_{i,j}$ is determined.

Again, this component is optional, and not every structure analyzer provides this functionality. For those structure analyzers which do not have this component, *ViStruclizer* simply takes $\rho_{i,j,u,v} = \pi_{i,u}\pi_{j,v}$ with the assumption that the probabilities of observing any particular edge in all edge clusters are the same.
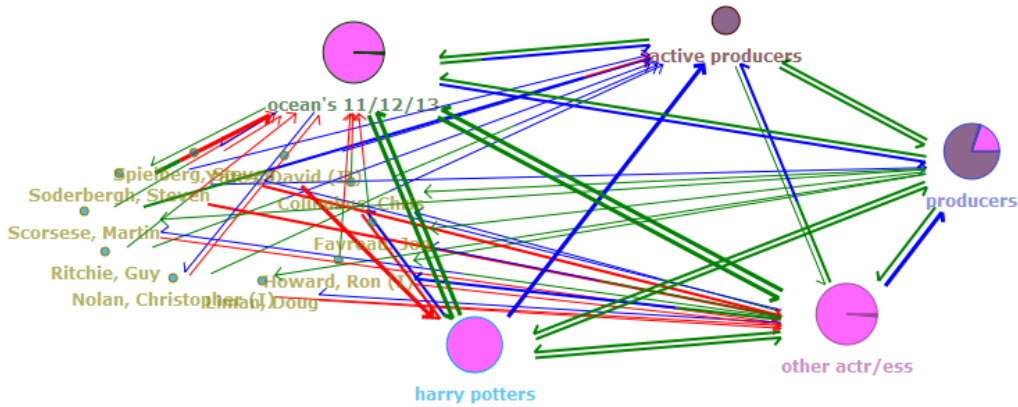
Fig. 4. The Structure of IMDb Network with `director` Cluster Expanded

3.2   Visual Representations

The summarized network structure of our IMDb network is shown in Figure 3.
Each vertex cluster is represented by a pie chart with a user-assigned cluster
label, size and pie segments determined by the weight and modal weights of the
vertex cluster respectively (as computed using Equation 1 and 2). Different colors
are assigned to different vertex modes, i.e., magenta for `actors/actresses`, royal
blue for mode `directors` and purple for `producers`.

There are two producer clusters, the active producers who produce relatively
more movies, and the other consisting of producers who may also act in the
movies, as shown by the magenta sector in the pie chart of the cluster `producers`
in Figure 3. Actors and actresses are divided into three clusters, i..e, "harry
potters", "ocean's 11/12/13", and "other actr/ess". The first and second groups
act in Harry Potter series and Ocean's series respectively, while the third group
includes the others. The royal blue pie chart represents the `directors` cluster.

Between two vertex clusters are edges representing edge clusters. The width of
the edge represents the weight of the corresponding edge cluster (the self-loops
on each vertex cluster are omitted), as computed by Equation 3. Three colors are
also used for three relations, green for `collaborate`, blue for `work_for` and red
for `direct`. The proportions of the three colors tell how the two vertex clusters
are related.

3.3   User Interactions in the IMDb Network

The visual representation of the original network and summary network in
ViStruclizer is meant to be interactive. Using the control panel at the bottom
of screen (see Figure 3), one can control the expansion threshold values to be
used, select the relations to be included in the visualization, search vertices, and
perform other operations on the network.

Figure 4 shows how *ViStruclizer* looks like after the "directors" cluster gets
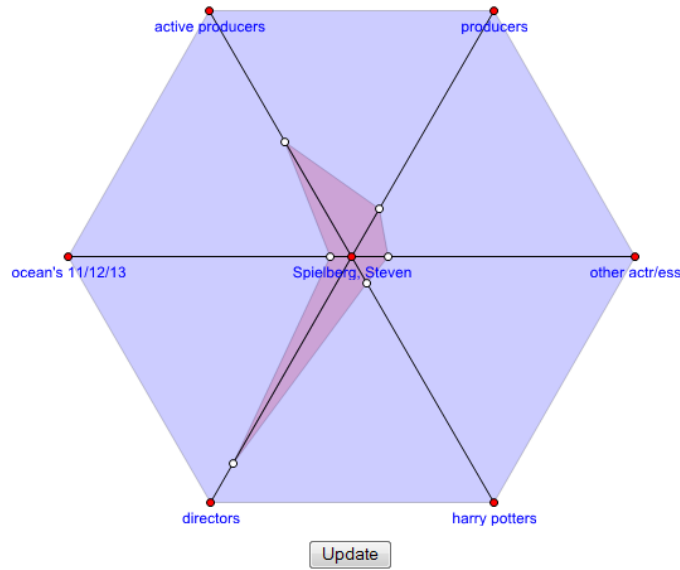expanded. An edge between the expanded vertex cluster and another vertex

Fig. 5. Updating of the Membership of a Vertex

cluster is broken down into edges between vertices and the vertex cluster. Similarly, the edge between a vertex and a vertex cluster is aggregated from edges linking this particular vertex and all vertices in the vertex cluster. By adjusting the expansion threshold, one can choose to only expand the vertices assigned to clusters with high membership probabilities. *ViStruclizer* also allows any selected vertex to be placed at the center of the screen to use it as the focus.

One can change the membership probabilities $\pi_{i,u}$ of a selected vertex V_i by invoking a membership refinement function. As shown in Figure 5, the membership probabilities of the selected vertex in different vertex clusters can be displayed in a radar chart. Each radial axis of [0,1] value range corresponds to a vertex cluster and one can choose any point along the axis. The membership probabilities $\pi_{i,u}$ will then be recomputed so that the sum remains 1, and the visualization of the vertex clusters and the edge clusters will altered accordingly.

4   Conclusion and Future Work

This paper presents *ViStruclizer*, a network visual analytics system designed and implemented based on a framework for visualizing multi-dimensional networks using their summary network structures. Multi-dimensional network models are new in social network analysis and there have not been many visualization techniques specially designed for them. *ViStruclizer* represents one of these pioneering efforts. With the incorporation of a structure analyzer, which performs positional and role analysis, *ViStruclizer* effectively allows users to explore a multi-dimensional network along with its summary network. Its visualization capabilities on an IMDb network have also been demonstrated. To carry this work further, we plan to improve the structure analyzer and visualizer components. In particular, other network models for multi-dimensional networks and

efficient learning of these models will be studied. For example, if we regard a specific topic as a type of relations, the Twitter[3] network can be visualized by different topics between clusters of users [6]. The visualizer component can also be improved by introducing new visual constructs that help user to identify interesting communities and anomalies in the networks.

References

1. Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P.: Mixed membership stochastic blockmodels. Journal of Machine Learning Research 9, 1981–2014 (2008)
2. Batagelj, V., Mrvar, A.: Pajek - Analysis and Visualization of Large Networks. In: Mutzel, P., Jünger, M., Leipert, S. (eds.) GD 2001. LNCS, vol. 2265, pp. 477–478. Springer, Heidelberg (2002)
3. Cao, N., Sun, J., Lin, Y.R., Gotz, D., Liu, S., Qu, H.: Facetatlas: Multifaceted visualization for rich text corpora. IEEE Transactions on Visualization and Computer Graphics 16(6), 1172–1181 (2010)
4. Contractor, N.S.: The emergence of multidimensional networks. Journal of Computer-Mediated Communication 14(3), 743–747 (2009)
5. Dai, B.T., Chua, F.C.T., Lim, E.P.: Structural analysis in multi-relational social networks. In: SDM, pp. 451–462 (2012)
6. Dai, B.T., Lim, E.P., Prasetyo, P.K.: Topic discovery from tweet replies. In: MLG: The Workshop on Mining and Learning with Graphs (2012)
7. Heer, J., Boyd, D.: Vizster: Visualizing online social networks. In: INFOVIS, p. 5 (2005)
8. Henry, N., Fekete, J.D.: Matrixexplorer: a dual-representation system to explore social networks. IEEE Transactions on Visualization and Computer Graphics 12(5), 677–684 (2006)
9. McLachlan, G.J., Krishnan, T.: The EM Algorithm and Extensions. Wiley Series in probability and Statistics. Wiley (2008)
10. Newman, M.E.J.: Fast algorithm for detecting community structure in networks. Physical Review E 69(6), 066133 (2004)
11. Shannon, R., Quigley, A.J., Nixon, P.: Graphemes: self-organizing shape-based clustered structures for network visualisations. In: CHI Extended Abstracts, pp. 4195–4200 (2010)
12. Shen, Z., Ma, K.L., Eliassi-Rad, T.: Visual analysis of large heterogeneous social networks by semantic and structural abstraction. IEEE Transactions on Visualization and Computer Graphics 12(6), 1427–1439 (2006)
13. Shi, L., Cao, N., Liu, S., Qian, W., Tan, L., Wang, G., Sun, J., Lin, C.Y.: Himap: Adaptive visualization of large-scale online social networks. In: PacificVis, pp. 41–48 (2009)
14. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press (1994)
15. White, H.C., Boorman, S.A., Breiger, R.L.: Social structure from multiple networks. i. blockmodels of roles and positions. The American Journal of Sociology 81(4), 730–780 (1976)
16. Zinsmaier, M., Brandes, U., Deussen, O., Strobelt, H.: Interactive level-of-detail rendering of large graphs. IEEE Transactions on Visualization and Computer Graphics 18(12), 2486–2495 (2012)

---

[3] `twitter.com`