NodeTrix-CommunityHierarchy: Techniques for Finding Hierarchical Communities for Visual Analytics of Small-world Networks

Jaya Sreevalsan-Nair and Shivam Agarwal
Graphics-Visualization-Computing Lab, International Instittue of Information Technology, Bangalore, India

Keywords: Small-world Networks, NodeTrix, Similarity Matrix, Hierarchical Communities, Workflow, Visual Analytics, Clustering Algorithm.

Abstract: While there are several visualizations of the small world networks (SWN), how does one find an appropriate set of visualizations and data analytic processes in a data science workflow? Hierarchical communities in SWN aid in managing and understanding the complex network better. To enable a visual analytics workflow to probe and uncover hierarchical communities, we propose to use both the network data and metadata (e.g. node and link attributes). Hence, we propose to use the network topology and node-similarity graph using metadata, for knowledge discovery. For the construction of a four-level hierarchy, we detect communities on both the network and the similarity graph, by using specific community detection at specific hierarchical level. We enable the flexibility of finding non-overlapping or overlapping communities, as leaf nodes, by using spectral clustering. We propose NodeTrix-CommunityHierarchy (NTCH), a set of visual analytic techniques for hierarchy construction, visual exploration and quantitative analysis of community detection results. We extend NodeTrix-Multiplex framework (Agarwal et al., 2017), which is for visual analytics of multilayer SWN, to probe hierarchical communities. We propose novel visualizations of overlapping and non-overlapping communities, which are integrated into the framework. We show preliminary results of our case-study of using NTCH on co-authorship networks.

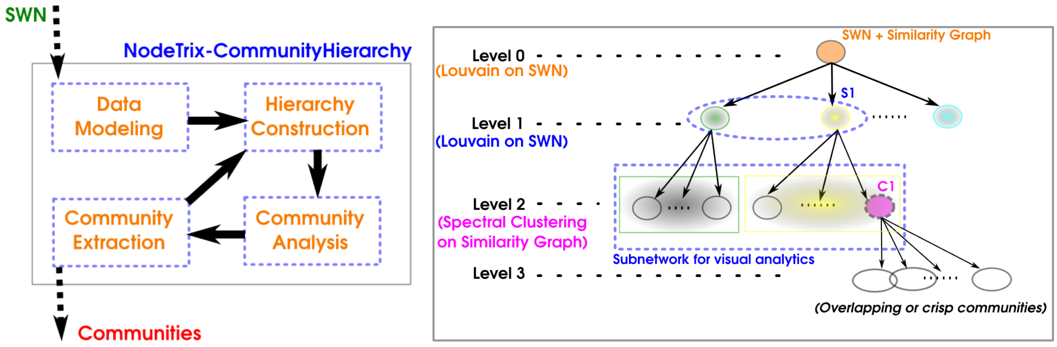
1 INTRODUCTION

Visual analytics of small world networks (SWNs), which include social networks, is an approach to extract knowledge from a complex network. Several existing visualizations of SWNs tend to exclusively use the data-space (Henry et al., 2007); while a small set of visualization techniques for multi-variate networks and multiplex networks make use of the metadata (i.e. node and link attributes) (Perer and Shneiderman, 2006) (van den Elzen and van Wijk, 2014). However, the question remains as to how much these visualizations help in fitting other data analytic processes into the data science workflow[1] of a network researcher or analyst.

Visual analysis of a large community becomes more tractable upon exploring its smaller *child* communities. Hence, hierarchical communities gives more insight to the dynamics of large networks. Both the network data and metadata can be used to probe and uncover such hierarchies. Here, we use node-similarity analysis for knowledge discovery from metadata. Use of visual analytics makes our targeted workflow semi-automated, with the domain expert-in-the-loop. Thus, we propose NodeTrix-CommunityHierarchy (NTCH), a set of techniques for visual analytics of hierarchical communities in SWNs. NTCH is designed to use nested views (Javed and Elmqvist, 2012) for compact visualizations; as well as, to use selective data and algorithms for building a four-level community hierarchy. Consider an instance of an outcome of NTCH – while the co-authorship network visualization uncovers information on locally dense subnetworks and their central actors, there is more knowledge that can be extracted from text analysis of abstracts of publications in the network. This information has the potential to demonstrate similarities in research profiles of authors, and further *predict* if two authors in a smaller community will publish together in future. Such localized information can eventually enable one to understand

[1] We disambiguate the usage of “workflow,” where our work refers to the analysis and reflection phases in the “research programming” workflow (Guo, 2012) or “data science” workflow (Guo, 2013), as opposed to scientific workflow systems (Davidson and Freire, 2008).

Sreevalsan-Nair J. and Agarwal S.
NodeTrix-CommunityHierarchy: Techniques for Finding Hierarchical Communities for Visual Analytics of Small-world Networks.
DOI: 10.5220/0006175701400151
In *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017)*, pages 140-151
ISBN: 978-989-758-228-8



(a) (b)

Figure 1: (a) Our proposed set of techniques, NodeTrix-CommunityHierarchy, for visual analytics of SWNs. (b) Schematic diagram of four-level community hierarchy in a SWN, constructed by using its metadata to generate the similarity graph and choosing nodes and community detection algorithms for further division.

the global dynamics of large networks. Another goal of NTCH is to explore the formation of overlapping communities, which is how real communities are formed. Overlapping communities is a challenge with respect to detection, representation, and visualization; due to which most of the existing work are limited to considering non-overlapping communities. Hence, NTCH has the flexibility of finding overlapping communities in the leaf nodes of the hierarchy, using spectral clustering.

We reuse the NodeTrix (Henry et al., 2007) for visualizing SWNs. NodeTrix exploits the "locally dense, globally sparse" topology of SWN, in providing a nested view in a hybrid visualization. Communities extracted using modularity-based methods, are locally dense subnetworks, which are represented as matrices or "aggregated nodes" in NodeTrix. These methods yield large communities in large SWNs. Network science has shown that a viable community must be of size 150 (the Dunbar number (Dunbar, 1998)), or more compactly, 100 (Leskovec et al., 2009). NTCH enables decision-making for community analytics, such as, *which communities* can be explored for further divisions and *which* community detection approaches can be used to find the leaf nodes (Figure 1(a)). Our previous work, NodeTrix-Multiplex (NTM) (Agarwal et al., 2017), is a visual analytic framework which extends NodeTrix with a focus+context approach for analyzing multiplex or multi-relational networks. Here, we use NTM to visualize SWN with its similarity graph/network layer, as well as to extend NTM to perform community analytics (Figure 1(b)).

Our novel contributions in NTCH are two-fold: firstly, in using a combination of visual analytics and quantitative analysis for making decisions on constructing a community hierarchy; and secondly, in extending NTM for cluster analytics on probing leaf node communities. We demonstrate preliminary results of using NTCH on two co-authorship networks.

Notations: A SWN is denoted as $\mathcal{N} = \{\mathcal{V}, \mathcal{E}, \mathcal{E}_S\}$, where $\mathcal{V}$ is the vertex[2] set of the network, and $\mathcal{E}$ the edge set, and $\mathcal{E}_S$ the edge set in the node-similarity graph. $e(u,v) \in \mathcal{E}$ or $\mathcal{E}_S$ is an edge exists between vertices $u, v \in \mathcal{V}$ and it stores edge weight, a normalized real value. $\mathcal{L}_i$ is the i^{th} level of community hierarchy of the network, and $C_j^{\mathcal{L}_i}$ is the j^{th} of the N_i communities in the i^{th} level (i.e., $0 \leq j < N_i$). $\mathcal{S}_i$ is the subnetwork of interest in the i^{th} layer, where $\mathcal{S}_i = \bigcup_k C_k^{\mathcal{L}_i}$, where k indicates selected communities. $C_j^{\mathcal{L}_i}$ and $\mathcal{S}_i$ are vertex sets; their edge sets contain edges whose vertices belong to the vertex sets, inclusively. In our work, In $\mathcal{L}_0$, $\mathcal{S}_0 = C_0^{\mathcal{L}_0} = \mathcal{N}$. N_i communities in $\mathcal{L}_i$ are detected when community detection is applied to $\mathcal{S}_{i-1}$. For nested community detection, we refer to N_c to be the number of communities that can be detected in a (generic) community C, irrespective of the hierarchical levels. For quantitative analysis, we use Newman-Girvan modularity as Q_h, generalized modularity as Q_g, silhouette coefficient as SC, and fuzzy partition coefficient as FPC. A density metric to check the "goodness" of the community detection within a selected subnetwork, R_e, is defined as the ratio of number of inter-community links to the total number of links in the subnetwork, prior to community detection. Intermediate matrices such as degree matrix, modularity matrix, weight matrix, cluster membership matrix, and identity matrix of size n are referred to as D, B, W, U, and I_n, respectively. The two co-authorship networks in our case-study are the IEEE Infovis conference (IV) and the IEEE VAST conference (VA) co-authorship networks.

[2] We refer to "network", "nodes" and "links" with respect to the dataset, and "graph", "vertices", and "edges," to the data structures, respectively.

2 RELATED WORK

We look at relevant work on visualization of communities in complex networks, and community detection techniques for finding overlapping communities in a hierarchy, which are integral parts of design decisions for NTCH.

Visualization of Communities in Complex Networks: NodeTrix (Henry et al., 2007) is a hybrid visualization of social networks, where the small world property of "globally sparse but locally dense" has been exploited to provide the layout. It integrates better readability of node-link and matrix representations of the network in respective scenarios (i.e. sparse and dense nature of the network which in the global and local spatial context, respectively) (Ghoniem et al., 2004). NodeTrix has been extended (Henry et al., 2008) to include node duplication to indicate overlap of a node in multiple communities. In our previous work on NodeTrix-Multiplex (NTM) (Agarwal et al., 2017), we use NodeTrix for the network visualization of multilayer SWNs. NTM introduces a focus+context approach by using communities in the SWN layer as foci. A hybrid data model is used in NTM, where any layer of the focus can be visualized; and the remaining network, i.e. the context, is visualized in another layer. NTM has used matrix seriation to finding patterns of near-cliques within a focus. In NTCH, we use these patterns to propose parameters for community detection within the focus. NTM enables users to find communities which persist across layers in these subnetworks. Our implementation of NTCH is built on the visual analytic tool developed using NTM. Similar to our proposed cluster visualization techniques, visualizations of groups in graphs (Vehlow et al., 2015) use logical visual groupings. In contrast to our matrix visualization techniques and nested views, node-link diagrams and integrated (linked) views have been widely used for visualizing hierarchical structures in networks (Rufiange et al., 2012; Shi et al., 2009; Vehlow et al., 2013). Detangler (Renoust et al., 2015) is a visual analytics system for multiplex networks, where new data abstractions, such as substrate and catalyst networks, have been used for visualization.

Hierarchical and Overlapping Communities in Complex Networks: The algorithms for identifying hierarchical overlapping communities in complex networks, often use agglomerative methods. In such methods, the overlap between communities is studied across layers. However, we use divisive methods using partitioning (clustering) methods, with a restriction on finding overlapping communities in $\mathcal{L}_2$ communities. The use of divisive methods and its restriction are due to the limitations of our proposed workflow in conjunction with use of visual analytics. In many of the existing agglomerative methods, each network node is added to multiple communities until a termination criterion is satisfied. This criterion is usually based on properties such as, node fitness (Lancichinetti et al., 2009), gain in similarity-based modularity (Huang et al., 2010), and local-first approach (Coscia et al., 2014). Divisive methods typically use Newman-Girvan modularity (Newman and Girvan, 2004), Q_h, as a termination condition for partitioning (Fortunato, 2010), e.g. Louvain community detection (Blondel et al., 2008), and yield non-overlapping communities. We have used the generalized modularity function, Q_g, as given in (Havens et al., 2013) for computing modularity for both overlapping as well as non-overlapping communities; Q_g being equivalent to Q_h in the latter.

Our use of similarity graph for analyzing the network is equivalent to an abstraction of a multi-relational or multiplex network (Kivelä et al., 2014). Use of modularity for finding non-overlapping (or crisp) communities has been extended to multilayer networks (Bennett et al., 2015)(Mucha et al., 2010). However, overlapping community detection in multilayer network has inherent challenges, e.g. percolation of communities across layers. (De Domenico et al., 2015) have proposed use of modular flows between nodes across layers to identify overlapping communities in multilayer networks, in flat hierarchy. We use a similar concept, by evaluating the modular flows occur in aggregated nodes (communities in $\mathcal{L}_2$) across layers in community hierarchy. Newman has proposed the use of spectral cuts using modularity matrix for community detection in networks (Newman, 2006) as an improvement over using the adjacency or weight matrix. In a similar vein, we propose to use spectral clustering for finding leaf node communities, with the flexibility of finding overlapping or non-overlapping communities.

Fuzzy c-means algorithm has been used for overlapping community detection in complex networks (Zhang et al., 2007; Xie et al., 2013). The soft modularity function Q_g (Havens et al., 2013), which is a generalized function for both crisp and fuzzy communities, has been an improvement over the modularity function given in (Zhang et al., 2007) for overlapping communities. Q_g gives probabilistic membership matrix whereas the latter uses possibilistic membership, with a user-defined threshold.

3 HIERARCHICAL COMMUNITIES

Different from NodeTrix, which is exclusively for visualizing the layout of SWNs, our motivation is to devise techniques for a "data science" workflow for exploring a community hierarchy in the network, using both the network data as well as the metadata. Two of the integral design decisions of our workflow is to perform network analysis for community hierarchy ; and incorporate processes which will allow finding the leaf node communities. For the former, we use the defining matrices of the network, such as adjacency and similarity; and for the latter, we use visual analytics of communities in the third level. Since our analysis is in the matrix space, matrix seriation is important for identifying interesting patterns in the matrix, needs to be included in our workflow.

Use of Metadata: Owing to the small world property, within two levels of community detection using modularity-based methods (e.g. Louvain), closely-knit communities are often uncovered in a SWN. Such communities are mostly complete subnetworks (near cliques), or subnetworks with hubs, owing to which further divisiveness in the community hierarchy using the network data causes *fragmentation*. In existing literature, use of community size as a parameter for finding the viability of a community has been established, using reference values of community size, such as, mean value of 8.4 (Huberman and Adamic, 2004), Dunbar number of 150 (Dunbar, 1998), or maximum size of 100 (Leskovec et al., 2009; Narasimhamurthy et al., 2010).

However, our hypothesis is that some of these communities are big ($\approx 30-100$) enough to further divide or "disintegrate" into smaller, *but relevant*, communities by using information from the metadata. Since the network data has been exhausted for generating two levels of the community hierarchy, we propose the use of metadata, specifically node and link attributes, to discover knowledge about the network, for finding leaf node communities. One such knowledge discovery method is the use of *a similarity matrix*, which has been in effective in visualization of a SWN (Parveen and Sreevalsan-Nair, 2013).

Similarity Graph: We transform the metadata of the network to a similarity matrix, thus effectively performing dimensionality reduction (Strehl and Ghosh, 2003). Similarity matrix is a square matrix of size n, computed using pairwise similarity scores between nodes, and it is the weighted adjacency matrix for the similarity graph. There are several algorithms in literature which use a combination of attributes from the links as well as the nodes for similarity computation (e.g., author-topic similarity graph (Rosen-Zvi et al., 2010) for co-authorship networks). *A similarity graph with* ε*-neighborhood* retains only those edges with weight (i.e., distance between the nodes connected by the edge) less than ε (Von Luxburg, 2007), for which we use a user-defined parameter. This makes the graph sparser than a fully connected graph, thus reduces the clutter in its matrix visualization. The generation of the similarity graph makes the SWN, a multi-relational or multiplex network. We use the network layer as structural layer and similarity graph/network layer as functional layer in NTM, as has been used in (Agarwal et al., 2017).

We use the similarity layer for finding the leaf node communities in the SWN. However, modularity-based methods, such as Louvain, will not work for mostly complete graph, such as the similarity graph. Hence, we propose *spectral clustering* for community detection in the similarity layer. In spectral clustering in networks, a network embedding in spectral space is determined, and the nodes are clustered using commonly used partitioning algorithms, such as k-means and fuzzy c-means (FCM). Spectral clustering gives us the flexibility to extract both overlapping and non-overlapping communities.

Matrix Seriation: Seriation is a process of sorting objects along rows and columns in a two-way one-mode matrix (e.g. adjacency, similarity, distance matrices) to identify pertinent patterns of clustering (Liiv, 2010). We visualize matrices automatically seriated using selected algorithms, namely visual assessment of clustering tendency (VAT) algorithm (Bezdek et al., 2007) and coarse seriation in CLUSION (Strehl and Ghosh, 2003). VAT uses the minimum spanning tree of the dissimilarity graph to give a sorted order of nodes, and upon reordering, the clusters appear as square blocks along the diagonal of the matrix. CLUSION uses a permutation matrix computed using the cluster membership matrix (Strehl and Ghosh, 2003), to group nodes in a cluster together. We use VAT to estimate number of clusters and CLUSION to display constituency of non-overlapping communities in the matrix. Auto-seriated similarity matrices gives effective visualization of the SWNs as well as its hierarchical clustering tendency (Parveen and Sreevalsan-Nair, 2013).

Spectral Clustering: Spectral clustering is done by applying partitioning algorithm (k-means, FCM, etc.) on the embedding of the network in spectral space. Spectral decomposition of the Laplacian of the weight (i.e. adjacency) matrix gives the embedding. We then perform normalized spectral clustering (Ng et al., 2002), where eigenvectors of the normalized Laplacian matrix form columns in the embedding matrix.

The normalized rows of the embedding matrix give the position coordinates of the nodes in the spectral space.The symmetric normalized Laplacian matrix, for a graph $G(V,E)$, of n vertices, degree matrix, D, and weight matrix, W, is given by: $L_{sym} = I_n - D^{-0.5}WD^{-0.5}$.

Spectral clustering can be done using either the normalized or the unnormalized Laplacian matrix. We choose to use the normalized Laplacian matrix L_{sym} because L_{sym} shows stronger and consistent convergence of spectral clustering algorithm (Von Luxburg, 2007). Hence, we propose to use the MULTICUT algorithm (Ng et al., 2002), which is a normalized spectral clustering algorithm that uses a normalized graph Laplacian. Zhang et al. (Zhang et al., 2007) have used spectral clustering using normalized graph Laplacian (random walk) $L_{rw} = D^{-1}W$, and FCM algorithm (Dunn, 1973) for finding overlapping communities in complex networks. Since we want to have a common spectral mapping leading to either partitioning algorithms (k-means or FCM), we use L_{sym} for the spectral mapping. Nonetheless, the eigenvalues and eigenvectors of both normalized graph Laplacians are related (Von Luxburg, 2007), and since the similarity graph without ε-neighborhood does not contain nodes with low degrees, both normalized graph Laplacians will give similar outcomes. At the same time, White et al. (White and Smyth, 2005) have used L_{rw} in order to maximize the modularity function Q_h (Newman and Girvan, 2004), which measures the quality of node clusters in a graph. Hence, we can explore the use of spectral mapping using L_{rw} in SWNs in NTCH, in future.

Hierarchical Approach: We propose a four-level community hierarchy for SWN analysis (Figure 1). We perform Louvain community detection twice on the SWN layer to obtain communities in $\mathcal{L}_1$ and $\mathcal{L}_2$. Popular methods based on modularity optimization, such as Louvain algorithm (Blondel et al., 2008), suffer from resolution limit (Fortunato and Barthelemy, 2007), which fails to identify communities in smaller networks, like the $\mathcal{L}_2$ communities. Hence, we use the similarity graph for each community and spectral clustering on it to get the leaf node communities. We choose spectral clustering using partitioning algorithms, so that, our approach has the flexibility of re-using the spectral embedding of the community for either k-means or FCM algorithms. This re-use makes the clustering computationally effective as spectral mapping is $O(n^3)$ for n nodes in the subnetwork. A point to note here is that, the use of FCM gives relative membership of a node across communities, but not a measure of overlap. Hence, the membership values of two nodes within a community cannot be compared.

We use a divisive hierarchical clustering method as opposed to agglomerative methods (Coscia et al., 2014), as we are interested in visually exploring the network and probing further into communities. Agglomerative methods are well-suited for finding which communities a specific node belongs to. However, even though neat layouts of the network, as in NT (Henry et al., 2007), can be achieved with either divisive or agglomerative methods, the former more efficient as the termination condition for building the network has more control. For the latter, the logical termination is when all nodes belong to a single cluster and few levels of hierarchy may still show more fragmented structure in comparison to the same number of levels of divisive hierarchy. Hence, we use a divisive method for performing visual analytics on a four- level community hierarchy. The entire network is at $\mathcal{L}_0$. Louvain community detection is applied $\mathcal{L}_0$ and $\mathcal{L}_1$ communities to get $\mathcal{L}_1$ and $\mathcal{L}_2$ ones, respectively. Spectral clustering, with user's choice of partitioning algorithm, on $\mathcal{L}_2$ communities gives the leaf node ($\mathcal{L}_3$) communities.

Adaptive Community Hierarchy: The objective of our work is to explore hierarchical communities in a SWN using visual analytics. Such an objective directs our proposed workflow towards allowing the user to make decisions on *which* communities to propagate the hierarchy further and *which* partition algorithms to use for leaf node communities. We provide users with sufficient information about the tendency of a community to form communities within itself. This information helps the user to "confirm" or "approve" further divisive clustering or community formation within a community, thus giving an *adaptive* community hierarchy.

We perform community detection in $\mathcal{L}_1$ and $\mathcal{L}_2$ communities, selectively. The rationale is if we blindly perform community detection in all communities, it leads to excessive fragmentation. Fragmentation causes a spike in the number of inter-community links, which causes clutter in the NodeTrix layout. The increase in clutter due to the excessive fragmentation causes the network to lose its "globally sparse" property. Thus, in order to avoid fragmentation, we "confirm" a $\mathcal{L}_1$ or $\mathcal{L}_2$ community C, for further division, based on its analytics. For $\mathcal{L}_1$, only if modularity Q_h of C is above a specific threshold, Q_h^T, *and* if R_e of C is as low as possible, Louvain algorithm can be applied on C. We can confirm only *after* performing the community detection and not a priori, because computing metrics of its community formation, such as Q_h and R_e. These metrics are needed to determine

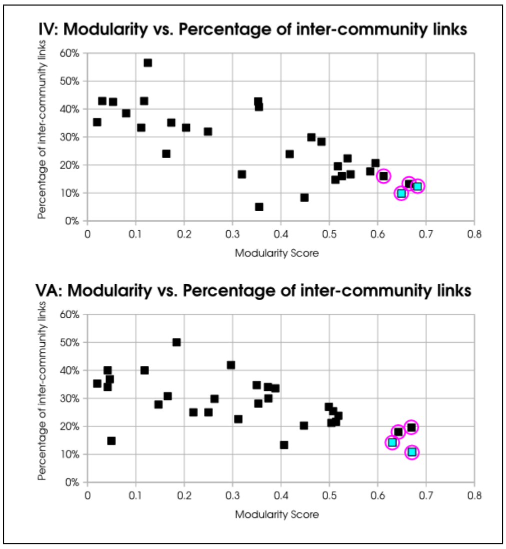


Figure 2: Q_h vs. R_e plots for selecting communities in $\mathcal{L}_1$ for further division using Louvain algorithm, in our case-study. Magenta highlights are communities with $Q_h > Q_h^T$ for a threshold $Q_h^T = 0.6$, amongst which cyan points are the ones with as low R_e as possible. Hence, the latter are selected.

the *goodness* of the community detection. Thus, analysis of the $Q_h - R_e$ relationship of $\mathcal{L}_1$ communities is used to select those for Louvain algorithm to find communities within themselves (Figure 2). Similarly, we selectively perform community detection within $\mathcal{L}_2$ communities of interest, which we determine by visualizing their VAT-seriated adjacency and similarity matrices to find interesting patterns. We allow the user to select the community detection method (spectral clustering with k-means or FCM) and confirm $\mathcal{L}_3$ communities, after considering the quantitative analysis and visualizations of the outcomes of the the chosen methods.

Semantics of Community Hierarchy: The semantics of the $\mathcal{L}_1$ and $\mathcal{L}_2$ communities are different from the $\mathcal{L}_3$ ones. The former are purely based on connected components or near cliques which are uncovered purely based on the relationship captured by the edges in the SWN, e.g. co-authorship relationship. The latter, on the other hand, captures the semantics of similarity within a community. A point to note here is that the similarity is computed from the information in the *metadata*, which is different from explicit information from the relationship captured by the edges. Hence, the semantics of the community hierarchy changes depending on the metadata analytics we perform. For instance, when using author-topic similarity to find the $\mathcal{L}_3$ communities in a co-authorship network, the $\mathcal{L}_3$ communities are formed by researchers who publish in similar topics. Even though it may seem trivially intuitive that co-authors in a $\mathcal{L}_2$ community would *definitely* work on topics of similar interests, it is not always true. When $\mathcal{L}_3$ communities are computed in the similarity space using author-topic similarity, the information encoded in the similarity graph is derived across all publications of such authors, including the ones they did not co-author. Hence, the authors in a $\mathcal{L}_2$ community may be connected in a near-clique, but could be working in diverse topics. One of the uses of such $\mathcal{L}_3$ communities is *link prediction*, i.e. find authors who have not co-authored, as per the data of the given network, but are similar. In the example, such authors are in the same community by virtue of their “connections” in the SWN and they have the potential of co-authoring papers, which may not be captured in the specific network, which may not be inclusive.

4 NodeTrix-CommunityHierarchy

We propose NodeTrix-CommunityHierarchy (NTCH), which is a set of techniques for visual analytics for SWNs, using hierarchical communities. NTCH enables users, such as network analysts, to make decisions on probing such communities, which are determined from the data as well as metadata of the SWN. NTCH uses specific user interactions (UIs) with communities; and community (or cluster) visualization techniques. For the former, the UIs are available in our previous visual analytic tool, NTM, and for the latter, we extend capabilities of NTM. Communities are represented using their adjacency matrices, which are visualized as aggregated nodes, as provided in the NodeTrix layout. We propose *UIs for spectral clustering as well as cluster visualization techniques* as an extension to NTM. Our proposed techniques are two different visualizations of the cluster membership matrix, U, using node-link as well as matrix representations. U is a rectangular matrix, which is an outcome of the partitioning algorithms, k-means or FCM. The rows and columns of U are clusters and nodes, respectively, and the matrix element is the normalized extent of membership of the node in a cluster. Cluster analytics in NTCH includes quantitative analysis of the communities in $\mathcal{L}_3$. The choice of using NodeTrix over node-link diagrams, e.g. as in Gephi (Bastian et al., 2009), is due to *clear separability* of the visualization of the community of interest, as a matrix, from the rest of the subnetwork in NodeTrix (Figure 3). This separability enables us to visually analyze any community represented as an aggregated node, and treated as a *focus* (Agarwal et al., 2017).

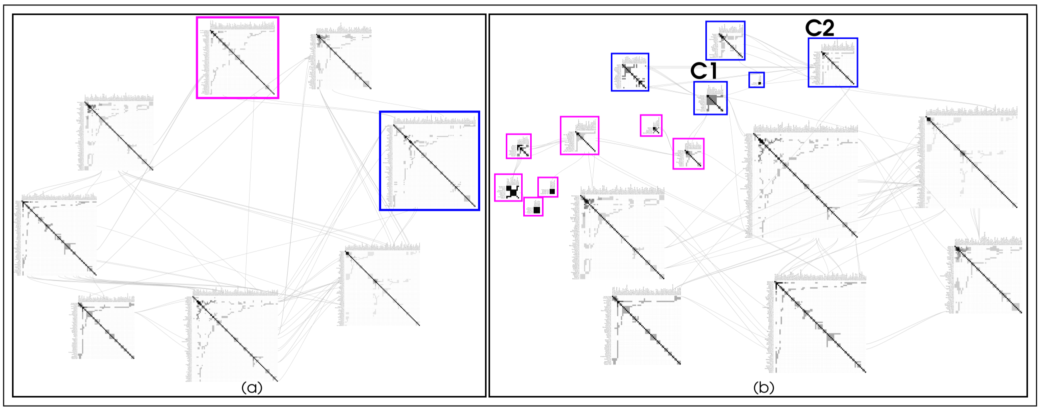


Figure 3: Visualizations of the IV network displaying communities in (a) $\mathcal{L}_1$, and (b) $\mathcal{L}_2$. The color coding shows the parent $\mathcal{L}_1$ communities of the corresponding $\mathcal{L}_2$ communities, obtained using Louvain algorithm. C1 (13 nodes, 37 intra-community edges), and C2 (26 nodes, 44 intra-community edges) show aggregated nodes, where Shneiderman and Heer are the central actors, respectively.

Aggregated Nodes: The aggregated nodes in NTCH are matrix representations of $\mathcal{L}_2$ communities, which are generated automatically based on constraints applied on $\mathcal{L}_1$ communities (Figure 2). The user can select one of the aggregated nodes as focus, using the focus+context approach in NTM; and perform spectral clustering on it. The choice of the partitioning algorithm (k-means or FCM) and parameters (e.g. number of clusters) are user inputs introduced in NTCH, for which the multi-layer visualization from NTM and VAT seriation are used. One of the noticeable differences between NodeTrix and NTM visualizations is that the diagonal of the unweighted adjacency matrices would have value 1 in the former, as opposed in 0 in the latter (colored as white and black, respectively, in grayscale colormap). This is because in NodeTrix, unweighted adjacency matrices are used, whereas we use weighted adjacency (or similarity) matrices and distance matrices for matrix visualization and spectral clustering, respectively. We compute distance matrices as difference of all-ones matrix and corresponding normalized weight matrix. Our visualization in NTM matches with that proposed in VAT and CLUSION.

Proposed Cluster Visualizations: In **cluster membership matrix representation**, U is rendered as a rectangular matrix using colormapping just like the square matrix of the aggregated nodes. Our proposed **cluster graph representation** is a node-link diagram, where both clusters and vertices are nodes of the diagram, which uses edge thickness to represent the membership value, u_{ij}. The cluster visualizations are currently included as an additional panel in the NTM tool.

Quantitative Analysis of Community Detection: We use metrics such as modularity, Q_g and cluster validity measures (silhouette coefficient and fuzzy partition coefficient), for quantifying the quality of community formation or clustering within a chosen community. We use Q_h for measurement of performance of Louvain community detection (on $\mathcal{L}_0$ and $\mathcal{L}_1$ communities). We use appropriate cluster validity measures for $\mathcal{L}_2$ communities for evaluating spectral clustering. For accommodating both non-overlapping as well as overlapping communities, we use a generalized modularity function (Havens et al., 2013), given by $Q_g = tr(UBU^T)/\|W\|$, where U is the $n \times N_c$ membership matrix for n nodes and N_c clusters/communities (overlapping or non-overlapping); modularity matrix $B = [W - \mathbf{m}^T\mathbf{m}/\|W\|]$; $\mathbf{m} = \{m_1, \ldots, m_n\}$, where $m_i = \sum_{j=1}^{n} w_{ij}$ and $\|W\| = \sum_{i,j=1}^{n} w_{ij}$. For non-overlapping communities, Q_g is equivalent to Q_h. Additionally, we compute quality metrics for partitions using cluster validity measures, such as, mean of silhouette coefficients of all nodes (Rousseeuw, 1987) for crisp partitions in k-means, and fuzzy partition coefficient (Pal and Bezdek, 1995) for fuzzy partitions in FCM.

Proposed Workflow: Here, we *stitch together* the design decisions discussed so far, i.e. the use of metadata, adaptive hierarchical community detection algorithm, and finding overlapping communities. Our workflow spans across the analysis and reflection phases in the research programming workflow (Guo, 2012). Guo describes these phases using action-level granularity; whereas we use process-level granularity. Our workflow consists of 4 stages (Figure 1): data modeling for analysis, hierarchy construction, community analysis, community extraction. In **data modeling**, we use a similarity function, appropriate for the application data, to generate a similarity matrix, i.e. $\mathcal{E}_S$ for the SWN. Between **hierarchy construction** and **community analysis**, we perform a community detection algorithm only on selected communities, based on qualitative as well as quantitative analyses of these communities. Upon "confirmation" of finding communities within communities, we perform **community extraction**, thus feeding back into **hierarchy construction**,

We introduce new UIs for implementing NTCH, for cluster analytics. Operations on aggregated nodes or foci include parameter selection for clustering, and cluster visualizations. In NTCH, the user can interactively choose parameters, such as, threshold for ε-neighborhood for similarity graph, seriation algorithm, clustering algorithm, and number of clusters. These additional UIs are supported in our Graphical User Interface (GUI) for NTM (Agarwal et al., 2017). **Subnetwork of Interest:** We have implemented our visual analytic tool for NTCH using D3.js library. Our tool is inclusive of all the UIs in NTM as well as new ones proposed here. We can load the entire network for the graph layout using NT, and use zoom capabilities in D3.js for visualizations. However, loading the entire network makes the UIs much slower. Hence, we load as many $\mathcal{L}_1$ communities as possible, as the application can accommodate for interactive speeds for loading and visualizing subnetwork containing ~ 500 nodes. We choose to load the $\mathcal{L}_1$ communities so that there is a logical grouping of nodes which are loaded together and analyzed further. The criteria for selecting $\mathcal{L}_1$ communities, we use here are based on its properties such as Q_h and N_c. The criteria we use are $Q_h > Q_h^T$ and N_c, where Q_h^T and N_c^T are user-defined thresholds, albeit are data-driven (Figure 4).

5 CASE-STUDY ON CO-AUTHORSHIP NETWORKS

Our case-study on co-authorship networks, uses the following datasets: Infovis (IV), and VAST (VA) co-authorship networks (Isenberg et al., 2015) during (1995-2015), and (2005-2015), respectively.

For **data modeling** in NTCH, we use the metadata, i.e. abstracts of papers used in the network data, to compute author-topic similarity (Rosen-Zvi et al., 2010). For **hierarchy construction**, we perform Louvain algorithm on the networks to obtain $\mathcal{L}_1$, and we get the results as shown in Table 1. We get N_1 communities in $\mathcal{L}_1$, however we select only N_1^* communities, which corresponds to subnetwork $\mathcal{S}_1$, to be loaded on NTCH. **Community analysis** enables selecting N_1^* communities (Figure 4), and two communities each in IV and VA networks for finding $\mathcal{L}_2$ communities (Figure 2). We further perform **community extraction** until $\mathcal{L}_2$ communities. On visual inspection, we select $\mathcal{L}_2$ communities whose central actors are: Shneiderman and Heer in IV, and Keim in VA, referred to as C1, C2, and C3, respectively (Figure 5)[3]. C1 has 13 nodes and 37 intra-community links; C2 has 26 and 4; and C3 has 100 and 475, respectively.

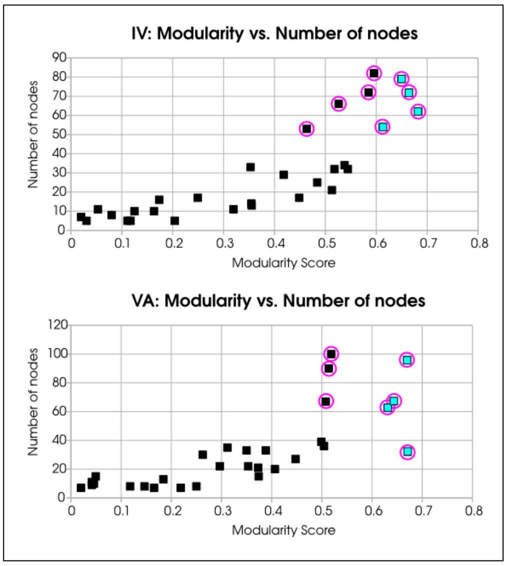


Figure 4: Q_h vs. N_c plots for selecting $\mathcal{L}_1$ communities in NTCH, in our case-study. Magenta highlights show communities which have $Q_h > Q_h^T$ and $N_c > N_c^T$, amongst which cyan points are those which satisfy the former exclusively. We use $N_c^T = 50$ and $Q_h^T = 0.6$.

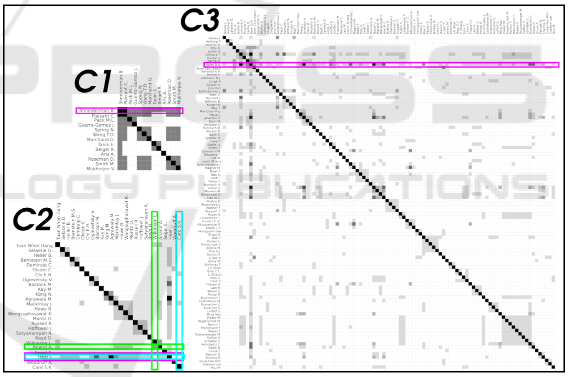


Figure 5: Aggregated nodes of C1, C2, C3 in the SWN, showing Shneiderman, Heer, and Keim, as central actors (magenta highlights), respectively. (Heer, Card) highlighted in cyan; (Anand, Wilkinson) in green.

We perform in-depth community analysis, which is specifically cluster analytics, on C1, C2, and C3, for finding $\mathcal{L}_3$ communities using the similarity graph. Louvain algorithm automatically gives 10, 7, and 8

[3] The images are better readable at high zoom levels (e.g. 400%), and higher resolution versions of the images are available at http://ntch.au-syd.mybluemix.net/

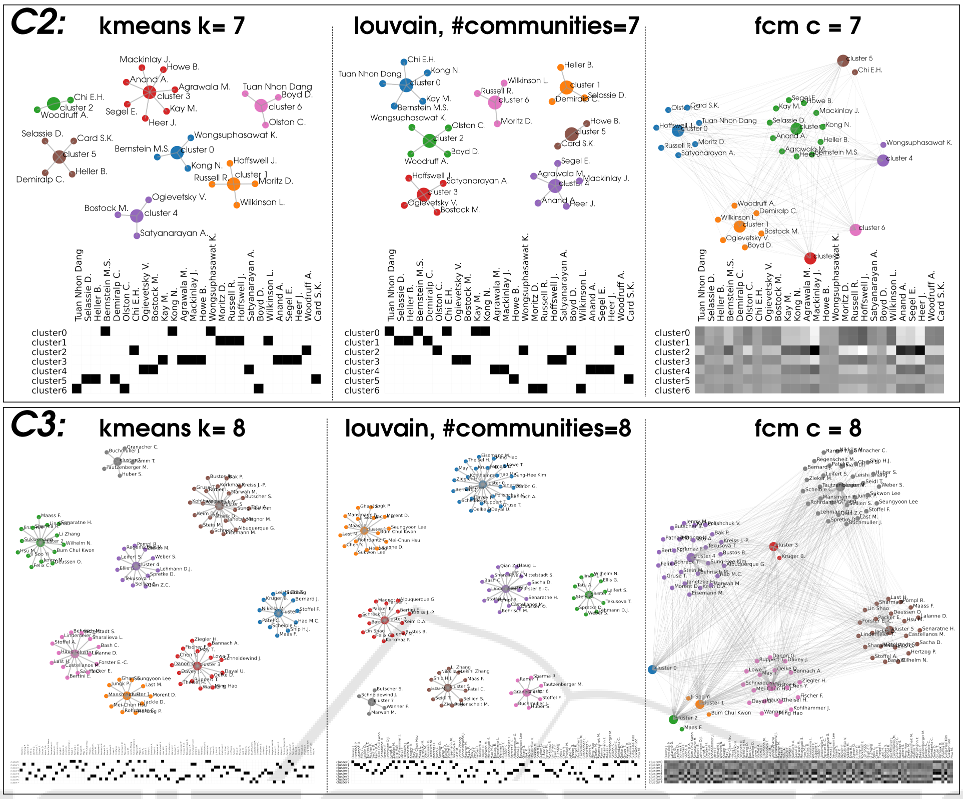


Figure 6: Cluster visualization for k=7 and k=8 clusters (or communities) for C2 and C3, respectively.

Table 1: Outcomes of number of communities in our case-study in $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2$. We perform Louvain algorithm on 2 communities each in $\mathcal{L}_1$ to get $N_2 = 18$ and 16 communities for IV and VA networks, respectively.

DS	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	N_1	N_1^*	$\mathcal{S}_1$	$\lvert\mathcal{E}(\mathcal{S}_1)\rvert$
IV	1235	2705	150	8	540	1318
VA	1266	3911	123	7	515	1862

communities in C1, C2, C3, respectively. We show both VAT and CLUSION seriations in C1-C3. Louvain algorithm gives 10 communities in C1, which has only 13 nodes, is excessive, which indicates that C1 inherently has poor edge density, which limits the performance of Louvain algorithm. The similarity matrix is mostly “homogeneous” (Figure 7), indicating weak community formation within C1, based on author-topic similarity.

Estimating Number of Clusters: Cluster analytics (Figure 6) gives 7 communities in C2, formed using k-means as well as Louvain, and overlapping communities using FCM for c=7. perform a similar analysis for 8 communities in C3. We make two observations – firstly, the results from Louvain and k-means partitions are not the same, owing to the difference in their optimization function; secondly, the FCM results show multiple empty clusters for C2 and fuzzy communities in C3, owing to dense inter-cluster links in the cluster membership graph visualization. Thus, this validates choices of user-defined parameters that when finding overlapping communities, analysis must be made on a lower number of clusters, in comparison to that of the non-overlapping communities.

We observe that FCM at lower number of clusters gives overlapping communities with a good balance of separability as well as overlap (Figure 8). The plots show variations in community detection outcomes using Louvain algorithm and spectral clustering (using both k-means and FCM). We see that Q_g is overall low for these communities, indicating that Q_g which is a metric based on edge density of the adjacency matrix, is not appropriate for distance-based measures of the similarity matrix. We have analyzed for a maximum of $\lceil\frac{|\mathcal{V}|}{3}\rceil$ for $\mathcal{V}$ nodes in the community. Q_g and SC values of Louvain algorithm are similar to the Q_g value of the corresponding k-means partitioning, at k=7 and k=8 in C2 and C3, respectively (Figure 8). This observation with respect to k-means and FCM partitioning confirms with the number of communities, which are detected by the Louvain algorithm. At these values of k, we also observe that the *FPC* due to FCM and Q_g due to k-means are co-incident with the values of Q_g and SC of the Louvain algorithm.

Improving FCM Results: We improve the FCM re-

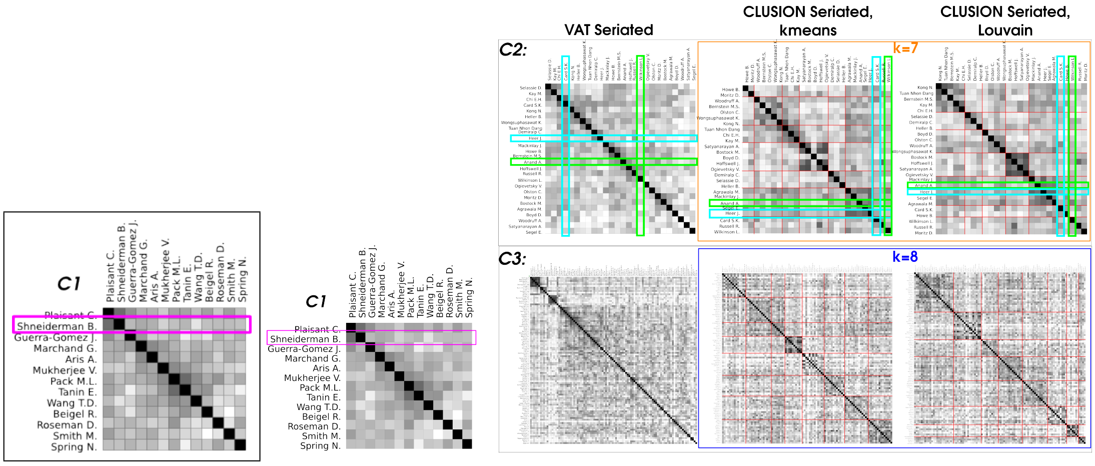


Figure 7: (left) VAT-seriated similarity matrix visualization of C1, (right) VAT- and CLUSION-seriated similarity matrix visualization C2 and C3. The latter shows Louvain and k-means clustering results for k=7 and k=8 clusters (or communities) for C2 and C3, respectively.

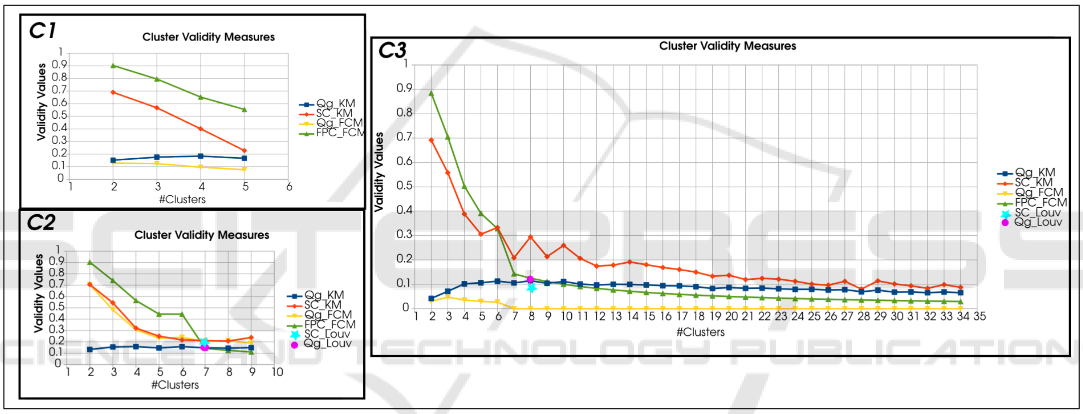


Figure 8: Quantitative analytics of modularity and cluster validity metrics for different number of communities/clusters, which are $\mathcal{L}_3$ communities.

sults by visualizing clusters for c=2 and c=3 for C2, and c=2 for C3. We find that C2 has more defined communities with good overlap, as opposed to C3. The difference in sizes of the 2 clusters in C3 indicates that the tendency to form communities based on author-topic similarity is comparatively low, as larger subset of the community belong to one cluster predominantly.

Insights About the Community and Network: We can gain insights such as link prediction and relevant overlap in communities, in a selected community using our proposed workflow. An example of link prediction is that in C2, Heer and Card do not have any IV papers, hence they do not have a link (Figure 5); but they are highly similar (Figure 7). Upon external investigation, we have found that {Heer, Card} have published in CHI and on other articles[4]. An example of a relevant overlap in communities, {Anand, Wikinson} fall in different communities (Figures 6 and 7), but have a strong inter-community link by virtue of having common papers (Figure 5). The strong inter-community link shows overlap between two communities. In NTCH, we visualize these communities in the context of a relevant larger subnetwork or the entire network, which enables on relationship of the authors outside their communities.

Expert User Evaluation: The data science workflow created using NTCH has been evaluated by a network science researcher. The expert has commented on the usefulness of such a workflow for a *mesoscopic* (community-based) analysis of a social network, by drilling down specific communities to enable further knowledge discovery. The expert has mentioned that the data model and the choice of processes includ-

[4] Heer, Jeffrey, Stuart K. Card, and James A. Landay. "Prefuse: a toolkit for interactive information visualization." In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 421-430. ACM, 2005.

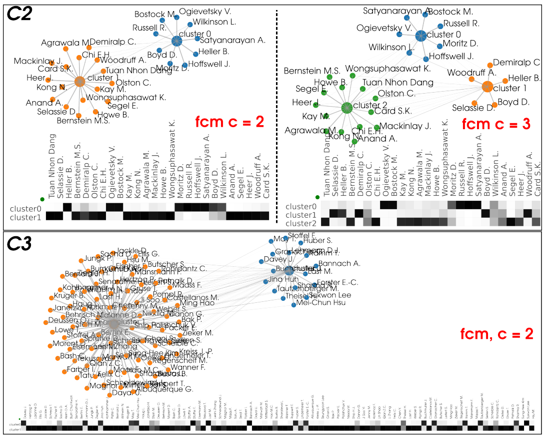


Figure 9: FCM visualization for lower values of k for C2 and C3.

ing the visualization make a meaningful workflow. The facility to perform cluster analytics on communities of size 100, such as C3, with supporting GUI, was found to be helpful, as real communities of this size are known to exist. However, the expert suggested improving the scalability of such a "locality-driven" workflow for studying "locally global" trends in larger parent communities, say in $\mathcal{L}_1$ communities in the community hierarchy.

6 CONCLUSIONS

In this paper, we have proposed techniques for visual analytics of a SWN, in a data science workflow, using hierarchical communities. Our proposed set of techniques is built on three core ideas, namely, using metadata in addition to network data for knowledge discovery, adaptive community hierarchy construction, and finding overlapping communities using visual analytics. While our workflow enables mesoscopic analysis of network in local scales, the design of the workflow has to be improved for analyzing larger parent communities. Our future work also includes analyzing other community detection algorithms for exploring overlapping communities. Currently, we focus on finding overlapping communities only in leaf nodes; however our workflow needs to be revised to finding overlapping communities across different levels in the community hierarchy.

ACKNOWLEDGEMENTS

The authors are grateful to Amit Tomar for initial implementations of the tool, and to the anonymous reviewers for comments in improving the paper. This work has been partially supported by funding from NRDMS, Department of Science & Technology, Government of India; RSA division of EMC^2 India; and INCOIS, Ministry of Earth Sciences, Government of India.

REFERENCES

Agarwal, S., Tomar, A., and Sreevalsan-Nair, J. (2017). *NodeTrix-Multiplex: Visual Analytics of Multiplex Small World Networks*, pages 579–591. Springer International Publishing, Cham.

Bastian, M., Heymann, S., Jacomy, M., et al. (2009). Gephi: an open source software for exploring and manipulating networks. *ICWSM*, 8:361–362.

Bennett, L., Kittas, A., Muirhead, G., Papageorgiou, L. G., and Tsoka, S. (2015). Detection of composite communities in multiplex biological networks. *Scientific reports*, 5.

Bezdek, J. C., Hathaway, R. J., and Huband, J. M. (2007). Visual assessment of clustering tendency for rectangular dissimilarity matrices. *Fuzzy Systems, IEEE Transactions on*, 15(5):890–903.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Coscia, M., Rossetti, G., Giannotti, F., and Pedreschi, D. (2014). Uncovering hierarchical and overlapping communities with a local-first approach. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(1):6.

Davidson, S. B. and Freire, J. (2008). Provenance and scientific workflows: challenges and opportunities. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1345–1350. ACM.

De Domenico, M., Lancichinetti, A., Arenas, A., and Rosvall, M. (2015). Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Physical Review X*, 5(1):011027.

Dunbar, R. (1998). *Grooming, gossip, and the evolution of language*. Harvard University Press.

Dunn, J. C. (1973). A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters.

Fortunato, S. (2010). Community detection in graphs. *Physics reports*, 486(3):75–174.

Fortunato, S. and Barthelemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41.

Ghoniem, M., Fekete, J.-D., and Castagliola, P. (2004). A comparison of the readability of graphs using node-link and matrix-based representations. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 17–24. Ieee.

Guo, P. (2013). Data science workflow: Overview and challenges. *Communications of the ACM*.

Guo, P. J. (2012). *Software tools to facilitate research programming*. PhD thesis, Stanford University.

Havens, T. C., Bezdek, J. C., Leckie, C., Ramamohanarao, K., and Palaniswami, M. (2013). A soft modularity function for detecting fuzzy communities in social networks. *Fuzzy Systems, IEEE Transactions on*, 21(6):1170–1175.

Henry, N., Bezerianos, A., and Fekete, J.-D. (2008). Improving the readability of clustered social networks using node duplication. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1317–1324.

Henry, N., Fekete, J.-D., and McGuffin, M. J. (2007). Nodetrix: a hybrid visualization of social networks. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1302–1309.

Huang, J., Sun, H., Han, J., Deng, H., Sun, Y., and Liu, Y. (2010). Shrink: a structural clustering algorithm for detecting hierarchical communities in networks. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 219–228. ACM.

Huberman, B. A. and Adamic, L. A. (2004). Information dynamics in the networked world. In *Complex networks*, pages 371–398. Springer.

Isenberg, P., Heimerl, F., Koch, S., Isenberg, T., Xu, P., Stolper, C., Sedlmair, M., Chen, J., Möller, T., and Stasko, J. (2015). Visualization publication dataset. Dataset: http://vispubdata.org/.

Javed, W. and Elmqvist, N. (2012). Exploring the design space of composite visualization. In *Visualization Symposium (PacificVis), 2012 IEEE Pacific*, pages 1–8. IEEE.

Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., and Porter, M. A. (2014). Multilayer networks. *Journal of complex networks*, 2(3):203–271.

Lancichinetti, A., Fortunato, S., and Kertész, J. (2009). Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015.

Leskovec, J., Lang, K. J., Dasgupta, A., and Mahoney, M. W. (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123.

Liiv, I. (2010). Seriation and matrix reordering methods: An historical overview. *Statistical analysis and data mining*, 3(2):70–91.

Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., and Onnela, J.-P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *science*, 328(5980):876–878.

Narasimhamurthy, A., Greene, D., Hurley, N., and Cunningham, P. (2010). Partitioning large networks without breaking communities. *Knowledge and information systems*, 25(2):345–369.

Newman, M. E. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104.

Newman, M. E. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113.

Ng, A. Y., Jordan, M. I., Weiss, Y., et al. (2002). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856.

Pal, N. R. and Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. *Fuzzy Systems, IEEE Transactions on*, 3(3):370–379.

Parveen, S. and Sreevalsan-Nair, J. (2013). Visualization of small world networks using similarity matrices. In *Big Data Analytics*, pages 151–170. Springer.

Perer, A. and Shneiderman, B. (2006). Balancing systematic and flexible exploration of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):693–700.

Renoust, B., Melançon, G., and Munzner, T. (2015). Detangler: Visual analytics for multiplex networks. In *Computer Graphics Forum*, volume 34, pages 321–330. Wiley Online Library.

Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P., and Steyvers, M. (2010). Learning author-topic models from text corpora. *ACM Transactions on Information Systems (TOIS)*, 28(1):4.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Rufiange, S., McGuffin, M. J., and Fuhrman, C. P. (2012). Treematrix: A hybrid visualization of compound graphs. In *Computer Graphics Forum*, volume 31, pages 89–101. Wiley Online Library.

Shi, L., Cao, N., Liu, S., Qian, W., Tan, L., Wang, G., Sun, J., and Lin, C.-Y. (2009). Himap: Adaptive visualization of large-scale online social networks. In *Visualization Symposium, 2009. PacificVis' 09. IEEE Pacific*, pages 41–48. IEEE.

Strehl, A. and Ghosh, J. (2003). Relationship-based clustering and visualization for high-dimensional data mining. *INFORMS Journal on Computing*, 15(2):208–230.

van den Elzen, S. and van Wijk, J. J. (2014). Multivariate network exploration and presentation: From detail to overview via selections and aggregations. *Visualization and Computer Graphics, IEEE Transactions on*, 20(12):2310–2319.

Vehlow, C., Beck, F., and Weiskopf, D. (2015). The state of the art in visualizing group structures in graphs. In *Eurographics Conference on Visualization (EuroVis)-STARs*, pages 21–40.

Vehlow, C., Reinhardt, T., and Weiskopf, D. (2013). Visualizing fuzzy overlapping communities in networks. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2486–2495.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

White, S. and Smyth, P. (2005). A spectral clustering approach to finding communities in graph. In *SDM*, volume 5, pages 76–84. SIAM.

Xie, J., Kelley, S., and Szymanski, B. K. (2013). Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)*, 45(4):43.

Zhang, S., Wang, R.-S., and Zhang, X.-S. (2007). Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A: Statistical Mechanics and its Applications*, 374(1):483–490.