APPLICATION OF MANYNETS TO ANALYZE DYNAMIC NETWORKS AND COMPARE ONLINE COMMUNITIES:

A CASE STUDY WITH NATION OF NEIGHBORS

Awalin Sopan
awalinnabila@gmail.com
Department of Computer Science, University of Maryland
December 2, 2011.

This scholarly paper is supervised by professor Ben Shneiderman and submitted to the Department of Computer Science, University of Maryland, College Park, MD 20742, USA in partial fulfillment of the requirements for the degree of Master of Science in Computer Science.

Abstract

Application of visual analytics for dynamic network analysis is a growing field of research nowadays. ManyNets is a network visualization tool that can visualize multiple network overviews at once and I participated in its development. I used this tool to analyze the dynamics of the online social network portal Nation of Neighbors (online communities for neighborhood crime watch) and made refinement to ManyNets in order to facilitate the analysis. In this report I demonstrate this gradual improvement process. A case study with Nation of Neighbors data shows the ability of ManyNets to glean insights from the users' activity log regarding the evolution and leadership in the communities. The study also suggests improvement of the tool and its interface, and future directions with this research.

TABLE OF CONTENTS

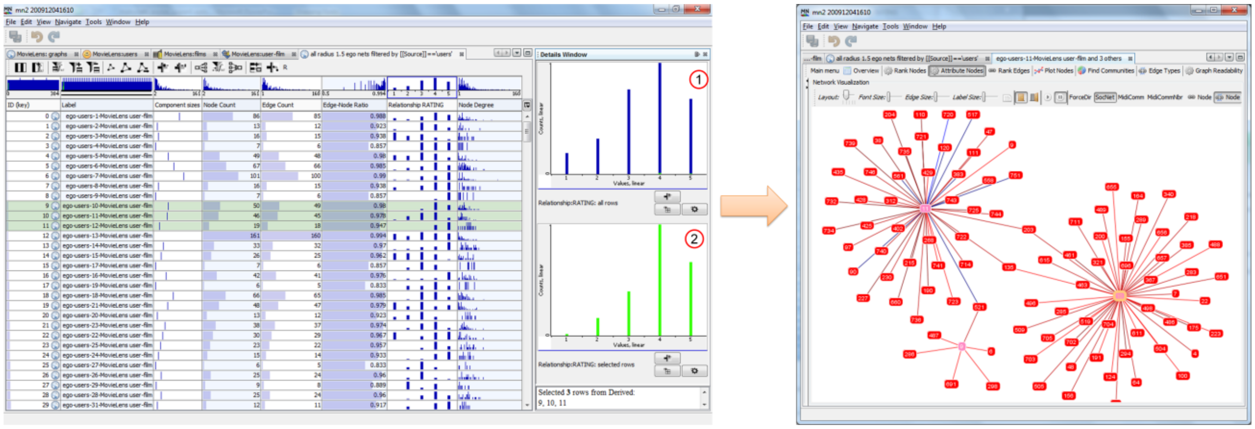
SECTION 1: INTRODUCTION

This scholarly report describes my work on developing ManyNets[15,30,31] and using it to analyze multiple networks from various domains; from movie recommendation system to temporal evolution of networks. ManyNets is a network visualization tool that can visualize multiple networks simultaneously. I am co-developing, designing and testing this software to analyze networks from different domains, namely: cell phone networks, movie recommendation system, user trust network, biological evolution network and at present the online community-based social network. This report presents the artifacts produced by the features I implemented in ManyNets and the analysis I performed using this tool. The major artifacts are:

- Distribution column overview:
    - Analysis of three datasets
    - Usability study
    - HCIL tech report
- Analysis of Online Communities:
    - Temporal evolution of online communities
    - Leadership identification among the communities

As the analysis depends mostly on the distribution column overview feature, I will explain it in detail in section 3. Before elaborating on the research, I will explain the development process involved with ManyNets. I have also documented a full user manual which is available online [19].

SECTION 2: MANYNETS: THE SOFTWARE



ID (key)	Label	Component sizes	Node Count	Edge Count	Edge-Node Ratio	Relationship RATING	Node Degree
0	ego-users-1-MovieLens user-film		86	85	0.988		
1	ego-users-2-MovieLens user-film		13	12	0.923		
2	ego-users-3-MovieLens user-film		16	15	0.938		
3	ego-users-4-MovieLens user-film		7	6	0.857		
4	ego-users-5-MovieLens user-film		49	48	0.98		
5	ego-users-6-MovieLens user-film		67	66	0.985		
6	ego-users-7-MovieLens user-film		101	100	0.99		
7	ego-users-8-MovieLens user-film		16	15	0.938		
8	ego-users-9-MovieLens user-film		7	6	0.857		
9	ego-users-10-MovieLens user-film		50	49	0.98		
10	ego-users-11-MovieLens user-film		46	45	0.978		
11	ego-users-12-MovieLens user-film		19	18	0.947		
12	ego-users-13-MovieLens user-film		161	160	0.994		
13	ego-users-14-MovieLens user-film		33	32	0.97		
14	ego-users-15-MovieLens user-film		26	25	0.962		
15	ego-users-17-MovieLens user-film		7	6	0.857		
16	ego-users-16-MovieLens user-film		42	41	0.976		
17	ego-users-19-MovieLens user-film		6	5	0.833		
18	ego-users-18-MovieLens user-film		66	65	0.985		
19	ego-users-21-MovieLens user-film		48	47	0.979		
20	ego-users-20-MovieLens user-film		13	12	0.923		
21	ego-users-23-MovieLens user-film		38	37	0.974		
22	ego-users-22-MovieLens user-film		30	29	0.967		
23	ego-users-25-MovieLens user-film		23	22	0.957		
24	ego-users-24-MovieLens user-film		15	14	0.933		
25	ego-users-27-MovieLens user-film		6	5	0.833		
26	ego-users-26-MovieLens user-film		25	24	0.96		
27	ego-users-29-MovieLens user-film		9	8	0.889		
28	ego-users-28-MovieLens user-film		25	24	0.96		
29	ego-users-31-MovieLens user-film		12	11	0.917		

Selected 3 rows from Derived:
9, 10, 11

Figure 1: ManyNets first look. Left) The MovieLens movie recommendation network split into user-film ego-networks, i.e. each row is a network with a user and the movies rated by him. Each user rates many films ranging from 1 to 5 stars. The networks' size, node count, edge count etc. are shown in columns. The distribution of ratings and node degree appear as histograms. The histogram overview of the whole Rating column for all the networks appears in the side pane (1) and the same for only the selected networks 9,10 and 11 is displayed just below it in green bars(2). Right) The merged node-link diagram of the selected networks 9,10 and 11. Pink nodes are users and red nodes are films rated by them.

SYSTEM ARCHITECTURE: ManyNets is designed under NetBeans platform to support multiple window manipulation. It uses Prefuse[25] for storing the data and rendering visualization for graph and table view. It has three basic modules:

- **ManyNets App**: This module acts as an application layer supporting the NetBeans api. It acts as a wrapper to use the library functions provided by ManyNets-lib.
- **ManyNets Lib**: This library provides the visualizations and data structures. It is pluggable to other java swing-based software. All the visualization, clustering, distribution and network manipulation code belong to this module. This module is reused in G-Pare [26] to generate the tabular view showing histograms inside a table.
- **Social Action-lib**: ManyNets uses SocialAction [16] code to show Node-Link diagram for the networks, so the SocialAction code is plugged into ManyNets using the SocialAction-lib module. SocialAction is another graph visualization tool integrated with ManyNets.

DATA STRUCTURE AND INPUT DATA FORMAT: ManyNets data-sets are composed of one or more schema descriptors and a series of text tab-delimited or comma-delimited table. A sample dataset may be organized as follows:

1**) Schema descriptor file in xml format**: A schema descriptor is an XML file that describes how the data tables relate to each other. It contains two parts.

- A description of the entities and relationships used in the dataset, in particular specifying which the key fields of each are, and which entities are connected by which relationships.

- A list of instances, each containing, for each entity or relationship in the previous section, a reference to a file (in tab-delimited or comma-delimited format) with the corresponding table.

Having multiple instances refer to the same schema allows loading multiple networks with the same general structure to be loaded and compared.

2) **Entity file(s):** for each instance, the entities associated with the instance that are mentioned in the schema file.

3) **Relationship file(s):** For each instance, the relationships associated with the instance that are mentioned in the schema file.

Once the data is loaded, several tables will appear. The rows of the tables may represent networks, nodes, edges, entities or relationships. For example, in MovieLens [6] dataset, users have users' rating movies, therefore two types of entities: 'films' and 'users'; and one type of relationship 'user-film-rating'.

Example config.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<config name="MovieLens">
  <description>    MovieLens dataset.
  </description>

  <schema>
    <entity id="films" idcol="ID" labelcol="Title"/>
    <entity id="users" idcol="ID" labelcol="ID"/>
    <relationship id="user-film-rating" idcol="ID"
       sourceid="users" sourcecol="ID1"
       targetid="films" targetcol="ID2"
    />
  </schema>

  <instance name="MovieLens">
    <table type="entity" ref="films" file="ml-movies.csv"/>
    <table type="entity" ref="users" file="ml-users.csv"/>
    <table type="relationship" ref="user-film" file="ml-um-movies.csv"/>
  </instance>
</config>
```

MANYNETS INTERFACE SUPPORTING NETWORK ANALYSIS: Each column in ManyNets table, regardless of table type, represents a statistic (for example, the number of links in a network, or the number of incoming edges for a node, or the age attribute of a user entity). At the top of each column header is a small column overview, which provides a summary of the contents of the column. ManyNets can handle integers, floating-point numbers, dates, strings, and distributions of these as data-types. Each network has its own node table and edge table and users can open these tables in separate window. Users can view each network as a node-link diagram generated by SocialAction in a separate tab. SocialAction's interface provides its own facilities for visualizing network overviews, zooming and filtering, and providing details on demand.

Subdividing a Network: ManyNets provides important ways to split a network into multiple sub-networks. These are as follows:

- Split by ego: Builds ego networks for each node in each network, and displays the results in a ManyNets window. Users can specify the radius of the ego.
- Split by cluster: Splits the network into clusters of connected components, retains/removes the largest cluster and removes the singleton components.
- Split by motif: Builds pairs and triplets by creating a new collection of networks, where each original network is subdivided into all the connected pairs or triplets that it contains.
- Split by feature: Splits the network into several sub-networks according to the values of a network feature and the user-specified interval of the value.

Filtering Nodes and Edges: Networks can be internally filtered by specifying python expressions that nodes and/or edges must (or must not) match. Users can choose either to 'Keep' or 'Remove' nodes and/or edges whose characteristics match the given expressions.

SECTION 2: DISTRIBUTION COLUMN OVERVIEW

Networks often have attributes distributed over a range, for example the degree distribution of nodes. Reducing the distributions to single-number statistics, such as median or average, is less informative than displaying the distribution itself. Additionally, distributions have specific properties such as skewness or bimodality which can be used to sort the column overviews, and similarity based algorithms can be used as well for clustering or sorting. In such cases, where an attribute can have values distributed over a range, comparing the distribution of values (in each of the aggregated units) can provide important insights. Therefore in ManyNets a new type of column, distribution column, is introduced where each cell in a column is a distribution of values. This distribution can be represented visually as histograms or heatmaps inside each cell. Tables often have too many rows to fit on the screen and scrolling is needed as the number of rows increases. If the data within a column exhibits a certain trend, or if two columns are correlated, users benefit from examining an overview of one or more columns. Previous works on table overviews only deal with columns where each cell contains a single value (a number or a string); here I address the problem introduced when cells contain distributions of values. The motivation to work on this problem came from the need to analyze trust ratings and movie ratings in a recommender system. For the analyst wanting to answer questions such as "do raters use the whole rating scale or not?" (i.e. looking at distributions of ratings per user) or "which films receive both very low and very high ratings?" (i.e. looking at distributions of ratings per movie), there was no tool available to browse or manipulate this data, to visualize and analyze a column of distributions. I implemented several designs for distribution column overviews and their interaction with the main table. I have used ManyNets as the platform to integrate this distribution overview. A table overview should allow analysts to "see the big picture", and identify clusters, trends and outliers that may be candidates for detailed inspection. Additionally, visual overviews can help to identify relationships between columns, such as possible correlation. Multiple display options and interactive sorting and filtering of the data are key aspects of the design of overviews. I have developed distribution overview as a part of ManyNets so that it can visualize the network attributes that are distribution of values. These distributions are created as the ManyNets aggregates the values of nodes and edges in network level. Later I also enabled ManyNets to read distribution type values directly from files; in that case ManyNets does not create the distributions rather they are directly read from input file. This way users can analyze distributions which are not part of any networks rather can be from any type of datasets.

For this work my research questions were:

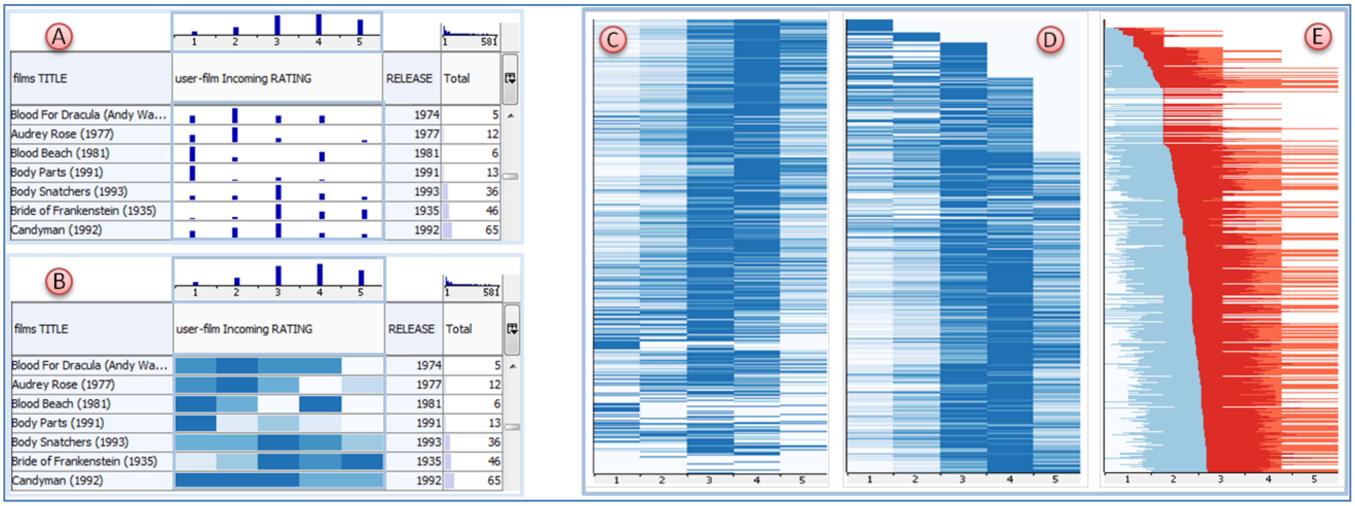
1) How to display the distribution overview in a way useful for analysis.

2) What interactions are required for analyzing the overviews.

I have analyzed four datasets using the distribution column overview in collaboration with Manuel Freire and designed and ran a usability study for the distribution overview technique with Meirav Taieb-Maimon. The datasets are from two movie recommendation systems, VAST 2008 mini-challenge cell phone network and US county population. The details of the analysis can be found in HCIL tech-report [28].

RELATED WORK: Early work on tables such as Table Lens [10] tackled the problem of overviews with a focus+context approach. InfoZoom [12] used a flipped overview with attributes located in the rows instead of columns, but it sacrifices the traditional table's basic property of having all attributes of a record aligned together [9]. Two popular stand-alone overviews for tables are heatmaps and parallel plots [7]. When rendering heatmaps, systems such as [4] and the Hierarchical Clustering Explorer [11](HCE) uses a dendogram to generate sorting for rows and/or columns. But in HCE, several overviews of the multivariate data are available, but they are all physically separated from the table. Kincaid's Line Graph Explorer [8] (LGE) uses a Table Lens-like interface to display line-graph data with a fisheye effect to reveal details. LGE uses color to provide a compact heatmap overview of the data. LGE also uses clustering to bring similar line-graphs closer together. However, in their naive dendogram ordering, the order of the child branches is essentially random. It uses a focus+context approach, and does not deal with the issues of linking multiple partial overviews on the same data. Additionally, LGE is intended to display a single column of linegraphs next to (but not inside) a table with traditional single-valued cells; line-graph columns are not intended to be freely mixed with traditional columns. However none of them considers distribution columns, neither do the general visual analytics packages Spotfire [2] and Tableau [1].

DISPLAYING THE OVERVIEWS: I implemented two approaches to build overviews of distribution columns: a) aggregated overviews that merge all the distributions into a single distribution (e.g. by summing all the bars and rescaling to fit) and b) row-based overviews attempt to draw compacted versions of all distributions at once. The column overview can be placed in a side panel by the table or in a separate resizable window.



films TITLE	user-film Incoming RATING	RELEASE	Total
Blood For Dracula (Andy Wa...		1974	5
Audrey Rose (1977)		1977	12
Blood Beach (1981)		1981	6
Body Parts (1991)		1991	13
Body Snatchers (1993)		1993	36
Bride of Frankenstein (1935)		1935	46
Candyman (1992)		1992	65

films TITLE	user-film Incoming RATING	RELEASE	Total
Blood For Dracula (Andy Wa...		1974	5
Audrey Rose (1977)		1977	12
Blood Beach (1981)		1981	6
Body Parts (1991)		1991	13
Body Snatchers (1993)		1993	36
Bride of Frankenstein (1935)		1935	46
Candyman (1992)		1992	65

Figure 2: On the left (i.e. A and B), a table containing distributions of rating received by movies along with total rating and release year. A) Rating distributions are presented as histograms inside table cells. B) Same table, now distributions are presented with heatmaps. Only part of the whole column can be shown and scrolling is required. The aggregated single cell histogram overview of all ratings is visible at the top of the column showing the global trend of the rating. On the right (C,D,E) are examples of compact row based overviews. Heatmaps can be used in the compact row based overview(C,D). In C, distributions are sorted by movie title but do not show any trend. D is sorted by highest rating and users can see most movies received highest possible rating 5 at least once. E shows stacked box-plot view, which is sorted by the average rating of the movie.

Row-based overviews transform and then compact each row of a distribution. For example, transforming the distribution histogram of each cell into a small heatmap-line, then reducing each row height to a single pixel results in a heatmap of the entire distribution column. Heatmaps are similar to histograms, but encode bin value as a color instead of the height of a bar. This allows "bars" to be shrunk to small, uniform heights. Then these heatmaps are stacked and this way they form larger heatmap overview, making fast visual comparison of adjacent heatmaps possible. In addition to heatmaps an additional possible transform is to use stacked boxplots to provide overviews of the most important statistics of each distributions; within each boxplot row, the maximum, minimum, average and a standard deviation above and below the average are encoded by color-field. Figure 2 explains the process of obtaining distribution overview.

To summarize, the following distribution overviews are supported:

1. **Histogram**: for aggregated distributions or aggregated non-distribution data; cannot be stacked.
2. **Heatmap:** built by stacking heatmaps of the component distributions.
3. **Box-plot**: built by stacking color-coded box-plots of the component distributions.

The row-based overviews can be generated in new window which is linked with the main table. As the table grows larger it is necessary to visualize the overview of the whole column. Heatmap overviews can facilitate this by visualizing a row-based overview. This window can be attached to the main table or can be redrawn in a separate window detached from the main table view. If the number of rows gets larger than the available pixel-height of the screen, ManyNets performs aggregation based on user-specified configuration; it can collapse similar rows to one, and renders the maximum, minimum or average value of the aggregated distributions.

To detect correlation among the columns the multi-column overview option was added which is able to draw overviews of several columns in the same window side by side and manipulate them as a group all together.

INTERACTING WITH THE OVERVIEW: For the analysis of distribution columns, users can choose the type of overview, the sort order for row-based overviews and also the distance metric (in case of similarity-based sorting and clustering). For histogram overview, users can choose appropriate scaling.

The main ordering options of the row-based heatmap overview are as follows:

1. **Sort by another column** of the table and then present them in that order in the overview, useful to reveal correlation of distribution pattern with other column. When an overview has been set to sort itself by an external source (another column in the table), it will mirror the order of the rows in that table, and update rendering whenever the table sorting changes.
2. **Sort by distribution descriptor** to reveal possible trend. For single values column users can sort by the value of the cell but distributions have different characteristics which users can use to sort the distributions. Therefore I have implemented distribution aware sorting using the value of the distribution descriptors. Users can sort the distributions using any of these descriptors. I have added the following statistics in the system:
    a. average,
    b. median,
    c. maximum,
    d. minimum,
    e. standard deviation,
    f. variance,
    g. skewness,
    h. kurtosis and
    i. bimodality.
3. **Sort by similarity** to find distributions that are similar to a given one and sort them according to that similarity measure. When sorting distributions with similarity-based algorithms, I compare whole distributions to one another to compute their pair-wise distance.
4. **Cluster similar distributions** together and order them according to dendogram sorting, useful for outlier detection and group detection. Cluster-based sorting of the distributions is performed using complete-linkage agglomerative clustering, with a second pass to rearrange the resulting dendogram using the optimal leaf ordering algorithm described in [3]. The resulting leaf order is then used as the sort order.

Comparing distributions: Users need to compare distributions to find similar ones or to generate clusters. To compare distributions I have used 4 metrics, these are:

1) Minimum Distance of Pair Assignment (MDPA)
2) Euclidean distance
3) Area under distribution
4) Kolmogorov-Smirnov distance

Distributions can have similar patterns as well as similar values or both. If users want to find distributions that are similar both in values and patterns, users need to compare both their shape and values, therefore compare them globally. To do this I used MDPA and Euclidean distance of similarity measure. If users only want to compare their shape, i.e., the relative proportion of the bins in each distribution, users should first normalize the values of each distribution and then compare. In this case the Area under the distribution and the Kolmogorov-Smirnov measure are used. The options are available through the interface.

Clustering distributions: For clustering the similar distributions, I have implemented dendogram sorting. First I have created the matrix of the distance/similarity of the pairs of distributions in the column, then ordered the leaves of the dendogram using the optimal leaf ordering algorithm. This approach takes relatively longer time, $O(n^3)$. Therefore another available option is to use Nearest Neighbor TSP algorithm which is not optimal but it uses heuristic to render the clustered overview faster, in $O(n^2)$ time. I have also tried using a genetic algorithm but the output was not deterministic and the process was slow, therefore I decided to discard that.

Interacting with main table: The distribution overview and the main table view are linked, therefore selection of rows in one view highlights rows in another view. After detection of outliers or clusters users can select the interesting distributions by dragging mouse in the overview panel and then the corresponding rows in the table will be also selected and highlighted. Then they can filter those rows thus creating subsets of the data, then generate new overviews from those subsets and compare them side by side. Figure 3 summarizes the interaction flow using distribution overview.

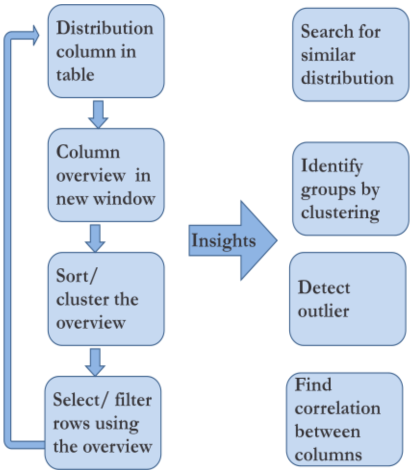

Distribution column in table
Column overview in new window
Sort/ cluster the overview
Select/ filter rows using the overview
Insights
Search for similar distribution
Identify groups by clustering
Detect outlier
Find correlation between columns

Figure 3: Workflow of using distribution overview

USABILITY STUDY: I designed a usability study where I have used the MovieLens dataset to train the participants and US county population dataset for the final evaluation. I prepared the script for training and evaluating the distribution overview, prepared questions and tasks for the experiment, set up the experiment and recorded the participants' activities during the experiment. After the usability study I reported the improvement suggestions from the participants and implemented some of them including the improvement of the column settings panel.

The next section presents several examples of how the column overview helped analyze online social network data.

SECTION 3: ANALYSIS OF ONLINE COMMUNITIES

General purpose social networking websites like Facebook and micro-blogging sites like Twitter have managed to have millions of active users whereas online social networks targeted towards communities with a specific goal are yet to thrive. For example, online social network for health-care, crime-watch or food-safety. One important research question is what makes this type of community successful and to analyze that users need to analyze their temporal evolution to see the elements that trigger the success/failure. Another goal is to analyze their growth over time that requires temporal analysis. Current network visualization tools are mainly focused towards visualizing the network structure of a single point in time leaving room for the design of tools that will help analyze temporal evolution of networks. I focus on using ManyNets to analyze the temporal growth of online communities, for this study I have used data from Nation of Neighbors (NON) [5] which is a platform for online neighborhood crime watch communities. I am considering the activities of its members to study the growth of the communities over time. While these communities are geographically distributed, they share the same purpose: sharing information and awareness about the safety of their neighborhood. By comparing and contrasting these communities users can select the interesting communities and what makes them distinct. For this study, I have worked with ManyNets and added new features that can help with temporal network analysis providing an interactive visualization. While analyzing the data, I had feedback from the user. There are new feature requests and improvement suggestions. I have prioritized these requests and implemented some of the important suggested features that are essential for the analysis of this dataset. For the assessment of community success, various metrics are defined and I have implemented them in ManyNets. In this study the user is P.J. Rey from the Department of Sociology, UMD. While his goal is to define metrics to quantify the success and growth of the communities, I have incorporated these metrics in ManyNets to support the exploration of the communities and compare them among themselves as well as analyze their growth over time.

The activities of the members are classified into 5 categories: post, reply, invitation, acceptance of invitation, reporting crime. All the activities are time stamped, so they are suitable for temporal analysis. Graduate student Jee-hye Kang annotated the conversation according to one of the possible topics, namely: community related, crime related, non related, safety related and others. I have examined what makes a community a successful one and the correlation between the success metrics and the features of the community such as age, activity of the users, involvement of Law and enforcement personals etc. to be more particular, the contributions are:

1. Examine the structural evolution of the networks: i.e., the activity [edges], the membership [nodes]. Two nodes share and edge if one of them replies to another one's post.

2. Examine the changes in the features of the node and edges of the network: for example the change of conversation topics over time. I have used the conversation and reports data posted in that site.

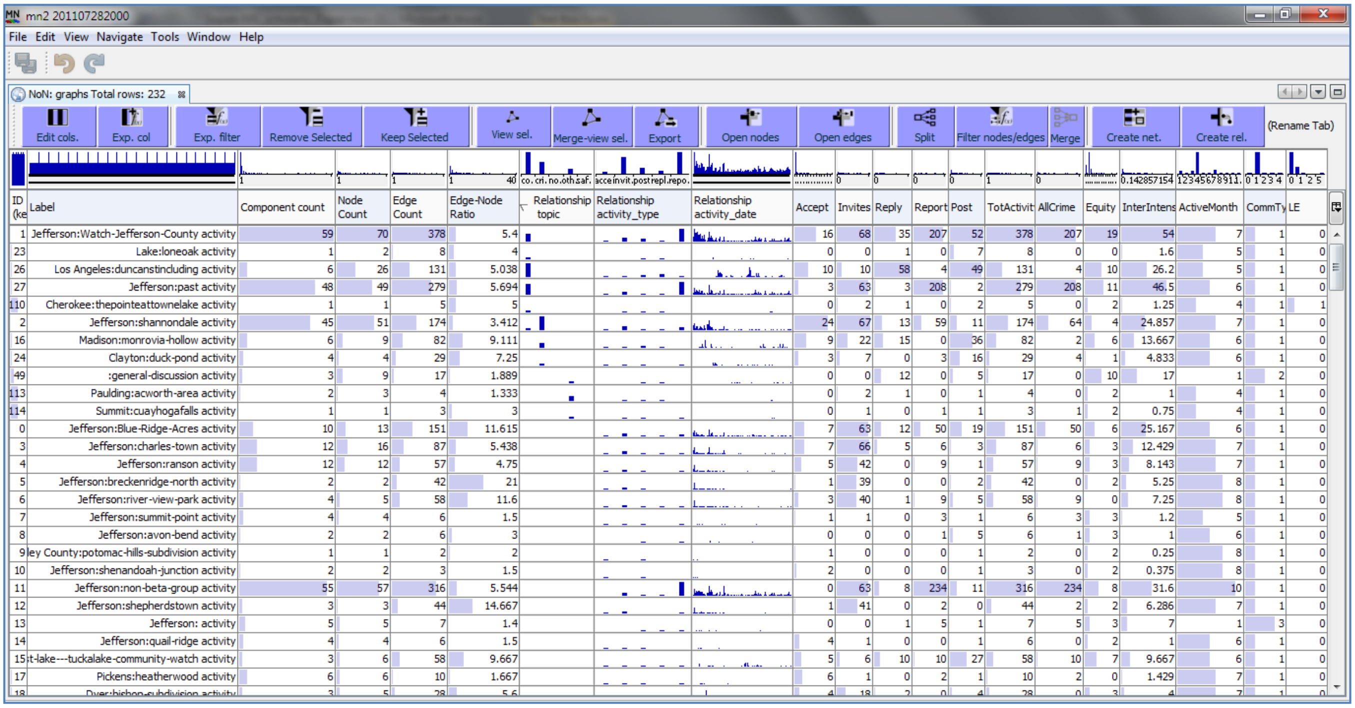
3. Establish a framework to identify successful and interesting communities using the metrics.

4. Identify the key influencing people in the communities. Characterize their activities and contributions.

5. Identify variables that may be correlated with the growth of the community; for example users can hypothesize that feature X is correlated with the overall activeness of the community. I developed features in ManyNets that facilitate the analyst to verify this hypothesis, for example, weather the presence of members from Law and Enforcement group helps the community to be more active or not.

6. Suggest feature requirements for visual analytics tools that can be used for exploration of temporal networks. These can be features from network level, node level and edge level.

7. Automate the process of analysis so that newer datasets can be loaded and analyzed easily.



ID (ke	Label	Component count	Node Count	Edge Count	Edge-Node Ratio	Relationship topic	Relationship activity_type	Relationship activity_date	Accept	Invites	Reply	Report	Post	TotActivit	AllCrime	Equity	InterIntens	ActiveMonth	CommTy	LE
1	Jefferson:Watch-Jefferson-County activity	59	70	378	5.4				16	68	35	207	52	378	207	19	54	7	1	0
23	Lake:loneoak activity	1	2	8	4				0	0	1	0	7	8	0	0	1.6	5	1	0
26	Los Angeles:duncanstincluding activity	6	26	131	5.038				10	10	58	4	49	131	4	10	26.2	5	1	0
27	Jefferson:past activity	48	49	279	5.694				3	63	3	208	2	279	208	11	46.5	6	1	0
110	Cherokee:thepointeattownelake activity	1	1	5	5				0	2	1	0	2	5	0	2	1.25	4	1	1
2	Jefferson:shannondale activity	45	51	174	3.412				24	67	13	59	11	174	64	4	24.857	7	1	0
16	Madison:monrovia-hollow activity	6	9	82	9.111				9	22	15	0	36	82	2	6	13.667	6	1	0
24	Clayton:duck-pond activity	4	4	29	7.25				3	7	0	3	16	29	4	1	4.833	6	1	0
49	:general-discussion activity	3	9	17	1.889				0	0	12	0	5	17	0	10	17	1	2	0
113	Paulding:acworth-area activity	2	3	4	1.333				0	2	1	0	1	4	0	2	1	4	1	0
114	Summit:cuayhogafalls activity	1	1	3	3				0	1	0	1	1	3	1	2	0.75	4	1	0
0	Jefferson:Blue-Ridge-Acres activity	10	13	151	11.615				7	63	12	50	19	151	50	6	25.167	6	1	0
3	Jefferson:charles-town activity	12	16	87	5.438				7	66	5	6	3	87	6	3	12.429	7	1	0
4	Jefferson:ranson activity	12	12	57	4.75				5	42	0	9	1	57	9	3	8.143	7	1	0
5	Jefferson:breckenridge-north activity	2	2	42	21				1	39	0	0	2	42	0	2	5.25	8	1	0
6	Jefferson:river-view-park activity	4	5	58	11.6				3	40	1	9	5	58	9	0	7.25	8	1	0
7	Jefferson:summit-point activity	4	4	6	1.5				1	1	0	3	1	6	3	3	1.2	5	1	0
8	Jefferson:avon-bend activity	2	2	6	3				0	0	0	1	5	6	1	3	1	6	1	0
9	ley County:potomac-hills-subdivision activity	1	1	2	2				1	0	0	0	1	2	0	2	0.25	8	1	0
10	Jefferson:shenandoah-junction activity	2	2	3	1.5				2	0	0	0	1	3	0	2	0.375	8	1	0
11	Jefferson:non-beta-group activity	55	57	316	5.544				0	63	8	234	11	316	234	8	31.6	10	1	0
12	Jefferson:shepherdstown activity	3	3	44	14.667				1	41	0	2	0	44	2	2	6.286	7	1	0
13	Jefferson: activity	5	5	7	1.4				0	0	1	5	1	7	5	3	7	1	3	0
14	Jefferson:quail-ridge activity	4	4	6	1.5				4	1	0	0	1	6	0	2	1	6	1	0
15	;t-lake---tuckalake-community-watch activity	3	6	58	9.667				5	6	10	10	27	58	10	7	9.667	6	1	0
17	Pickens:heatherwood activity	6	6	10	1.667				6	1	0	2	1	10	2	0	1.429	7	1	0
18	Dyer:bishop-subdivision activity	3	5	28	5.6				4	18	2	0	4	28	0	3	4	7	1	0

Figure 4: The network table of NON, each row is a community. For each community, the columns show basic network statistics, then the distribution of topic in the conversations, the distribution of the activity type, the temporal distribution of activity, and then total counts for each of the activity in separate columns (Accept, Invites, Reply, Report and Post), Total activity, AllCrime (post and report relating to crime), Equity, Interaction Intensity, Average Active Month , Type of community, and LE (number of members from Law Enforcement).

After users have the data in proper format users used that to analyze in ManyNets. In this process I implemented new features in ManyNets required for the analysis; the most important ones are:

1. Temporal Split.
2. Addition of new metrics.

Through ManyNets the timestamps of the activities are also aggregated and made into a distribution column which gives the overview of the temporal aspect of the activities. A heatmap overview of the temporal activity log is presented. This way users can observe which community is talking about which topics and if there is any shift of topic over time for a single community.

Distribution overviews in NON data: I have analyzed the following distribution columns along with their heatmap overviews:

1. Type of activity in the community: the activities are distributed among 5 types: invites, accept, post, reply, report.
2. Topic classification for each community: the topics of the conversations are coded into following types: non, community, safety, crime and others.
3. Tone of the conversation: the possible tones are negative, positive or neutral.
4. Tone of the speaker: positive or negative.
5. Activity date: the date of the activities distributed over time. This column is useful in particular to observe the temporal evolution of the communities.

RELATED WORK: To visualize dynamic networks two common approaches are a) plotting summary statistics over time [13,19] and b) presenting a separate node-link diagram of the network at each point of time, for example Powel et el[23]. The arrival of nodes and edges can be shown using different colors. But a different structure would cause the node and edges positions to be changed causing cognitive load to understand the change. Durant et al. [14] presented a snapshot-based network visualization showing different node positions over time. The “movie” approach is used for dynamic visualization. Moody [21] distinguished 1) Flipbook style movies where node-positions were fixed but connectivity formation was captured and 2) Dynamic movies where node-positions changed over time. Using sliding time frame to animate the network is introduced in Condor (or TeCFlow) [17,18]. Animation approaches might distract users to track changes in the network, for example to track new nodes, possibly responsible for significant change in the network, as nodes and edges keep changing their positions. Trier[24] used node degree as a measure of inertia so the high degree nodes move less across time and the dynamic movie becomes less distracting. TempoVis [27] and TimeSpring [29] are recent works dealing with dynamic network visualization. TempoVis keeps the node positions unchanged assigning colors to new coming nodes and TimeSpring uses a new layout algorithm so the nodes in adjacent timeframes are not too far apart. In contrast, ManyNets uses tabular visualization to compare features of networks; it can be same network in different time or different networks aggregated over time. Even though the network structure can be visualized on it, the main emphasis is given on the features of the networks and how they can be sliced. Identifying the relevant data for online communities is as important as visualizing it. Preece [32] describes relevant measures that needed to be analyzed for online community success. Butler [33] presents a model of online social structure that helps further to identify the key variables to analyze. After defining these measures, analysts can come up with hypotheses and verify them using various features of ManyNets.

DATA IMPORT: The SQL dump from NON is stored in our local MySQL database in several tables. MySQL database contain the tables: member, group,group_memebership, activity, invitation. Here the group corresponds to a community, each member belongs to one or more groups, and activity table contains the activity type, timestamp and the user id. As our purpose is to analyze the data as a network, edges are created from this database using SQL queries.

The database does not contain the topics of the conversation; these topics come from XL sheet provided by our collaborator graduate student Jee-hye Kang also provided hand-coded types for the tone of the conversations and the topics of the conversations. I merged these two datasets into one and imported in ManyNets. Hence the ManyNets network table shows the topic classification and the activity classification as distribution columns. Therefore a python script is run to integrate the topics with the query result from MySQL database to combine all the data and create the final dataset for analysis.

Finally I created different edge files for each community and load the edge files in ManyNets, from these files, ManyNets creates network and aggregated metrics for the communities and the nodes. In ManyNets tables the nodes are the members of the communities and the edges are their activities.

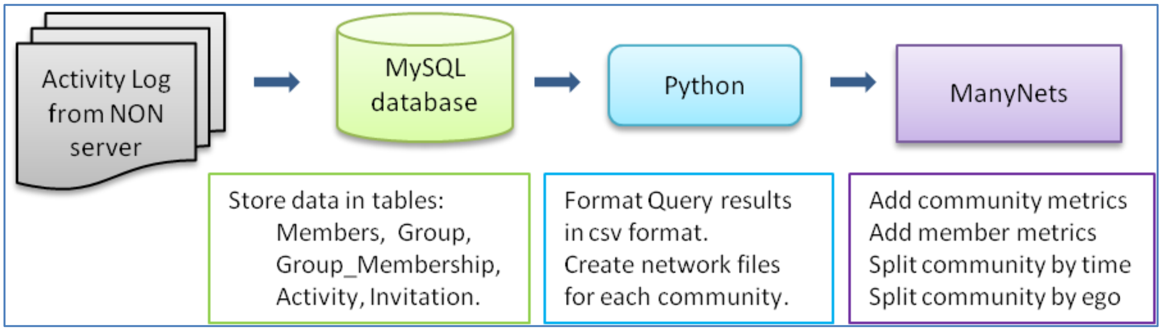
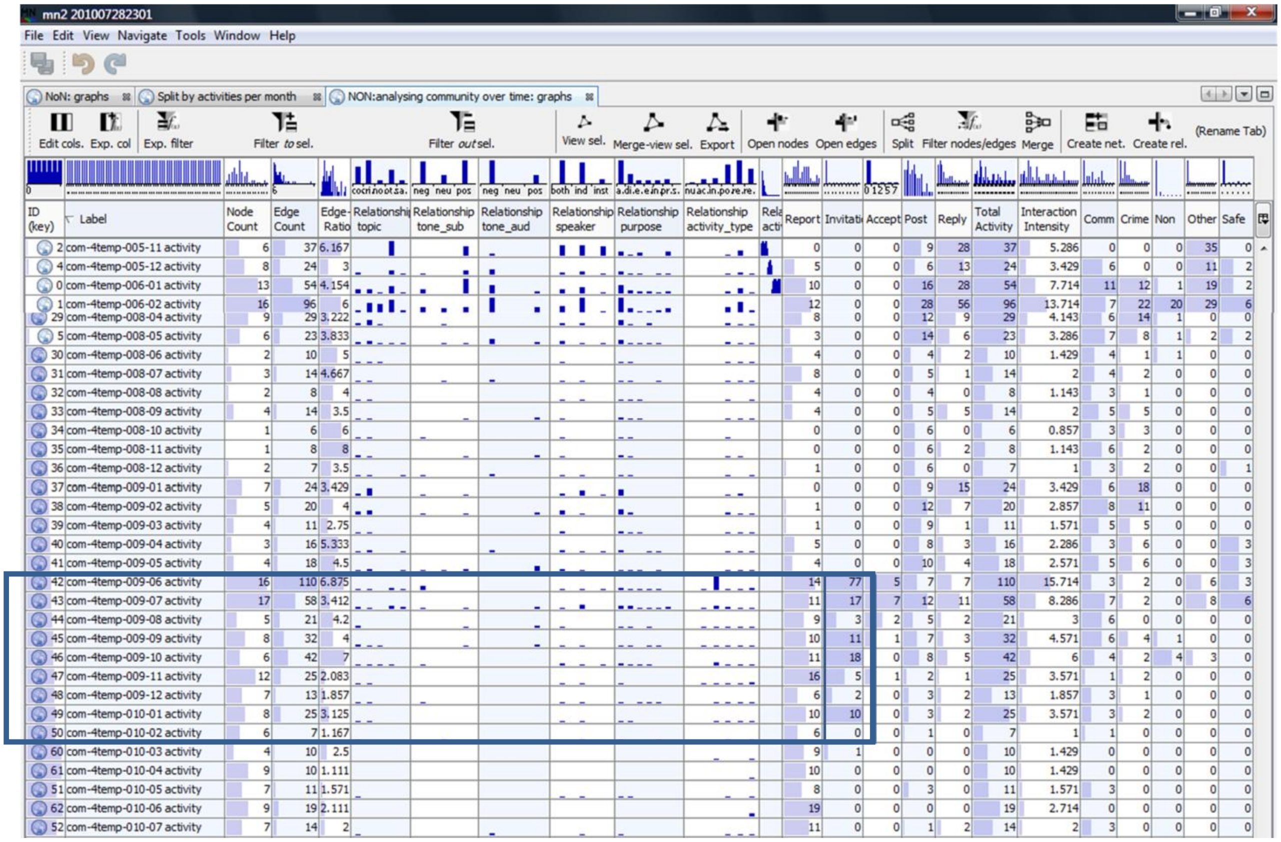


Figure 5: The dataflow from NON server to ManyNets table

DATA CLEAN UP: A moderate portion of time was spent on formatting and cleaning up the data and the use of ManyNets was helpful regarding the data validation. The provided data is the activity log of NON users from 2005 to 2011 which is a good duration of time for the analysis of the network evolution. As Watch Jefferson County was the most active community I chose this one to analyze its evolution and split this network over time, a separate network for each month. After this split, each row is the aggregation of activity for a month. I found the following discrepancies:

1. After June 2006, I observe a large number of invitations whereas there was no invitation activity before that (Figure 6). So it seemed interesting to know what caused this huge number of invitations at that time. Later I came to know that the site was redesigned and new feature was added after June 2006. The invitation activities before that time period were not recorded and the number shown in that month is actually the total invitations up to June 2006, not only for that month. As the site started with new features after this time, users decided to include only the data after June 2006 as comparing the activities before that would create misleading conclusions.



ID (key)	Label	Node Count	Edge Count	Edge-Ratio	Relationship topic	Relationship tone_sub	Relationship tone_aud	Relationship speaker	Relationship purpose	Relationship activity_type	Rela acti	Report	Invitati	Accept	Post	Reply	Total Activity	Interaction Intensity	Comm	Crime	Non	Other	Safe
2	com-4temp-005-11 activity	6	37	6.167								0	0	0	9	28	37	5.286	0	0	0	35	0
4	com-4temp-005-12 activity	8	24	3								5	0	0	6	13	24	3.429	6	0	0	11	2
0	com-4temp-006-01 activity	13	54	4.154								10	0	0	16	28	54	7.714	11	12	1	19	2
1	com-4temp-006-02 activity	16	96	6								12	0	0	28	56	96	13.714	7	22	20	29	6
29	com-4temp-008-04 activity	9	29	3.222								8	0	0	12	9	29	4.143	6	14	1	0	0
5	com-4temp-008-05 activity	6	23	3.833								3	0	0	14	6	23	3.286	7	8	1	2	2
30	com-4temp-008-06 activity	2	10	5								4	0	0	4	2	10	1.429	4	1	1	0	0
31	com-4temp-008-07 activity	3	14	4.667								8	0	0	5	1	14	2	4	2	0	0	0
32	com-4temp-008-08 activity	2	8	4								4	0	0	4	0	8	1.143	3	1	0	0	0
33	com-4temp-008-09 activity	4	14	3.5								4	0	0	5	5	14	2	5	5	0	0	0
34	com-4temp-008-10 activity	1	6	6								0	0	0	6	0	6	0.857	3	3	0	0	0
35	com-4temp-008-11 activity	1	8	8								0	0	0	6	2	8	1.143	6	2	0	0	0
36	com-4temp-008-12 activity	2	7	3.5								1	0	0	6	0	7	1	3	2	0	0	1
37	com-4temp-009-01 activity	7	24	3.429								0	0	0	9	15	24	3.429	6	18	0	0	0
38	com-4temp-009-02 activity	5	20	4								1	0	0	12	7	20	2.857	8	11	0	0	0
39	com-4temp-009-03 activity	4	11	2.75								1	0	0	9	1	11	1.571	5	5	0	0	0
40	com-4temp-009-04 activity	3	16	5.333								5	0	0	8	3	16	2.286	3	6	0	0	3
41	com-4temp-009-05 activity	4	18	4.5								4	0	0	10	4	18	2.571	5	6	0	0	3
42	com-4temp-009-06 activity	16	110	6.875								14	77	5	7	7	110	15.714	3	2	0	6	3
43	com-4temp-009-07 activity	17	58	3.412								11	17	7	12	11	58	8.286	7	2	0	8	6
44	com-4temp-009-08 activity	5	21	4.2								9	3	2	5	2	21	3	6	0	0	0	0
45	com-4temp-009-09 activity	8	32	4								10	11	1	7	3	32	4.571	6	4	1	0	0
46	com-4temp-009-10 activity	6	42	7								11	18	0	8	5	42	6	4	2	4	3	0
47	com-4temp-009-11 activity	12	25	2.083								16	5	1	2	1	25	3.571	1	2	0	0	0
48	com-4temp-009-12 activity	7	13	1.857								6	2	0	3	2	13	1.857	3	1	0	0	0
49	com-4temp-010-01 activity	8	25	3.125								10	10	0	3	2	25	3.571	3	2	0	0	0
50	com-4temp-010-02 activity	6	7	1.167								6	0	0	1	0	7	1	1	0	0	0	0
60	com-4temp-010-03 activity	4	10	2.5								9	1	0	0	0	10	1.429	0	0	0	0	0
61	com-4temp-010-04 activity	9	10	1.111								10	0	0	0	0	10	1.429	0	0	0	0	0
51	com-4temp-010-05 activity	7	11	1.571								8	0	0	3	0	11	1.571	3	0	0	0	0
62	com-4temp-010-06 activity	9	19	2.111								19	0	0	0	0	19	2.714	0	0	0	0	0
52	com-4temp-010-07 activity	7	14	2								11	0	0	1	2	14	2	3	0	0	0	0

Figure 6: The table shows the discrepancy about Invitation activity, no Invitation before June 2009, then sudden increase of Invitation.

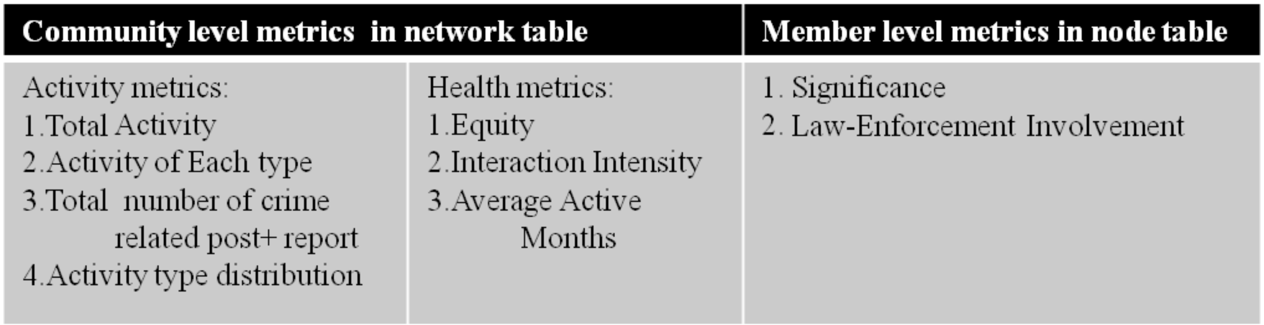
2. In the node table, after adding the metric for leadership, I observe that the member with id 0 is the most active one for most of the communities; therefore the initial conclusion was this member is a leader. But after discussion with the manager of this online community manager, it was confirmed that there is no member with id 0, rather whenever a person posts anonymously, he/she is stored in the database as member 0. Therefore I recalculated the activity measures for the members, their average activity, and significance of activeness and standard deviation of activity.

NEW FEATURES IN MANYNETS: For this analysis I implemented several new features in ManyNets which are described in this section.

1) Temporal Split: Temporal split of network in ManyNets produce several sub-networks where each sub-network contains activities of a month. Given any network, split it into network segments including a specific time period and then having each segment in a row of ManyNets network table, therefore users can sort the table by time and visualize the activity histogram of each network segment, this way they can see if there is any spike of activity in particular month. And also users can see the distribution overview of the types of activities to learn if any specific type of activity increased or decreased over time.

2) Dynamic Scaling of Distribution Values: When comparing the communities, it is important to measure cross community activities. From the miniature histogram in each distribution cell, users can see which activity prevails in each community. It is also important to observe if the total number of that particular activity is significant enough or not. Therefore ManyNets includes another functionality to dynamically change the scaling of histogram bar height inside the cell. Now users can specify if they want local or global scaling. Local scaling is important if they want to see the relative frequencies of the activities for each community, global scaling is used when users want to see the relative frequency of a particular activity type in global context.

3) Addition of new metrics: To analyze this dataset, I added several new metrics [Table 1] to network table and node table that are important to measure the community growth and to identify the leaders in the communities.



Community level metrics in network table		Member level metrics in node table
Activity metrics: 1.Total Activity 2.Activity of Each type 3.Total number of crime related post+ report 4.Activity type distribution	Health metrics: 1.Equity 2.Interaction Intensity 3.Average Active Months	1. Significance 2. Law-Enforcement Involvement

Table 1: The metrics added in ManyNets to analyze the communities

COMMUNITY LEVEL METRICS: These activity and health metrics are measured for each community and shown in the network table.

Activity Metrics are added as both an aggregated distribution column and separated columns. These are: total reports, posts, replies, invites, acceptance, and the summation of all of them. As all reports are also crime related, I also added other statistics summing over crime reports and crime related conversations.

- A_C = Communication Activity = $\sum$ (reports + posts + replies)
- A_I = Invitation Activity = $\sum$ (invites sent + invites accepted)
- $\sum_{(A)}$ = Total Activity = $\sum (A_C + A_I)$
- $\sum_{(UM)}$ = Total User-Months = $\sum$ (months since each user registered)
- M_{bar} = Mean Months Active Per User = $\sum_{(UM)} / \sum_{(U)}$
- Ratio = number of activity with the topic Crime / total number of activities
- All Crimes= All reports + all posts and reply that have the topic related to ‘Crime’ according to the topic classification.

The Health metrics are as follows:

a) Equity
b) Interaction Intensity
c) Average Active Months

Also for each community, now the table shows the number of Law Enforcement people in that community, so far analysts have not found any correlation with the activeness of the community with the involvement of LE personnel.

a) Equity: the variance in activity per month per user. Equity can be measured as a function of reciprocity of communication across ties. Because most communications are broadcast because users tend to make new posts rather than reply, analysts cannot use average number of two-way ties between users. Instead users propose to look at the variance in activity per month per user.

- E = Equity = $s_{(}A_{C)}^{2}$

b) Interaction Intensity: This is the total activity divided by total user-months.

- I = Interaction Intensity = $\sum_{(A)} / \sum_{(UM)}$

MEMBER LEVEL METRICS: The aim of these metrics is to identify leaders for each community. From the member table, analysts can observe the activity of each user, and then sort the users by their total number of activity, this way they can identify the most active users who are influencing the whole community dynamics. Only looking at the total activity is not always useful, therefore I have added another metric for finding leaders in the network, this metric is called Significance which is positive if the person is a leader and negative or zero if his/her activeness is not significant in the community. This metric is added in the Node table, but can be aggregated in the network table and from that users can see if a network has such nodes and how many of them. This is useful to measure the correlation of the growth of a community with the number of leaders in that community. Also the tone of the speaker towards the audience is encoded and shown in the table.

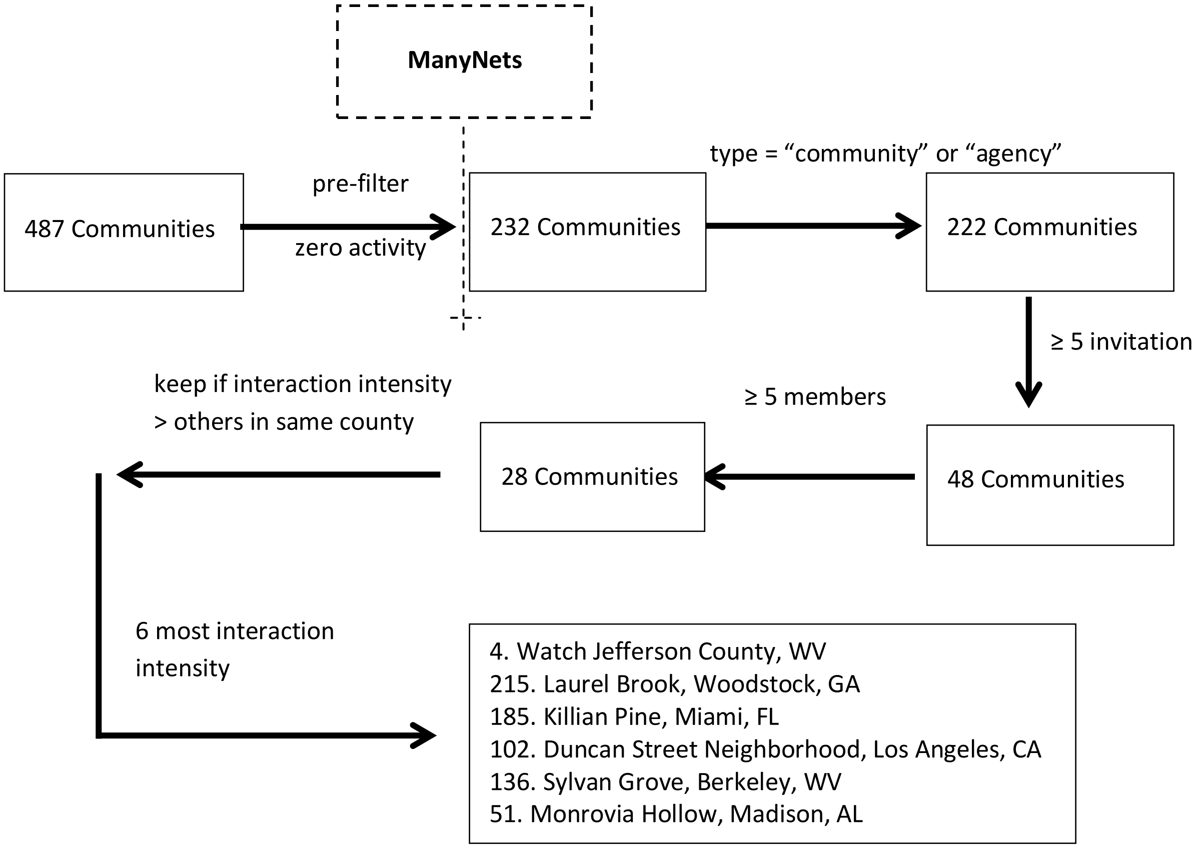
- Significance: Total activity count of the member- (standard deviation of activity count * 2 + average activity)

Sampling Procedure to Identify Important Communities: The analysis started with sampling of the communities. Using the metrics implemented in ManyNets P.J. Rey proposed the following framework to sample the interesting communities:

1) Filter out communities with no activities.
2) Then keep only the communities or agencies from the dataset.
3) Keep only the communities that have at least 5 invitation activities.
4) Among these communities, keep only the ones having at least 5 active members.

5) Then sort by Interaction Intensity and keep the 6 communities with highest Interaction Intensity.

Using this procedure it is possible to identify the most active communities for further analysis.



ManyNets
487 Communities
pre-filter
zero activity
232 Communities
type = "community" or "agency"
222 Communities
≥ 5 invitation
48 Communities
≥ 5 members
28 Communities
keep if interaction intensity
> others in same county
6 most interaction
intensity
4. Watch Jefferson County, WV
215. Laurel Brook, Woodstock, GA
185. Killian Pine, Miami, FL
102. Duncan Street Neighborhood, Los Angeles, CA
136. Sylvan Grove, Berkeley, WV
51. Monrovia Hollow, Madison, AL

Figure 7: Framework for sampling interesting communities.

DATA ANALYSIS AND INSIGHTS:

Loading al the communities in ManyNets gives an overview of the activity types and the growth of the communities. The first view of the community table is shown in Figure 4.

Activity Pattern: To get an overview of the global activity pattern among the 232 communities, I observed the aggregated histogram of the activity type column, it shows that over all, invitation is the most prevalent activity and then comes reports (Figure 8a) . After filtering down only to the larger communities (the 47 communities with at least 5 members) then it shows a different pattern (Figure 8b), in this case there are more reports than invitations.

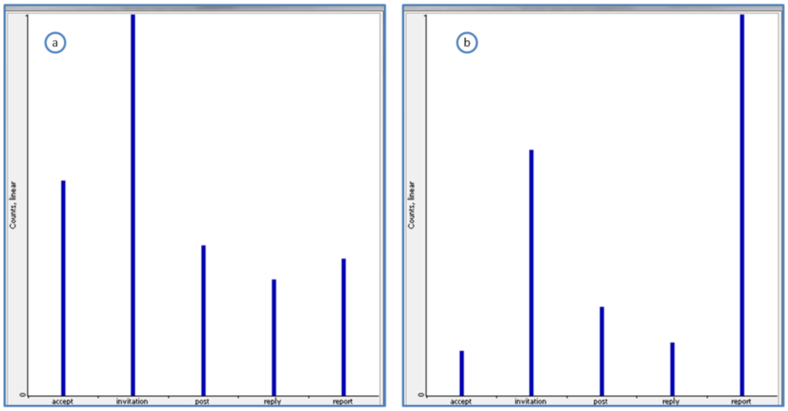


Figure 8: a) Overall activities in all communities: more invitations than any other activity. b) Overall activity in highly active communities: more reports than invitations.

To see if there is correlation among the size of the communities and the activity patterns, I used the same 47 communities. I have used the multicolumn overview for this having the node count column and activity_type distribution column side by side, sorting the rows according to node count. This shows that the larger communities have more reports than any other activity (Figure 9) .

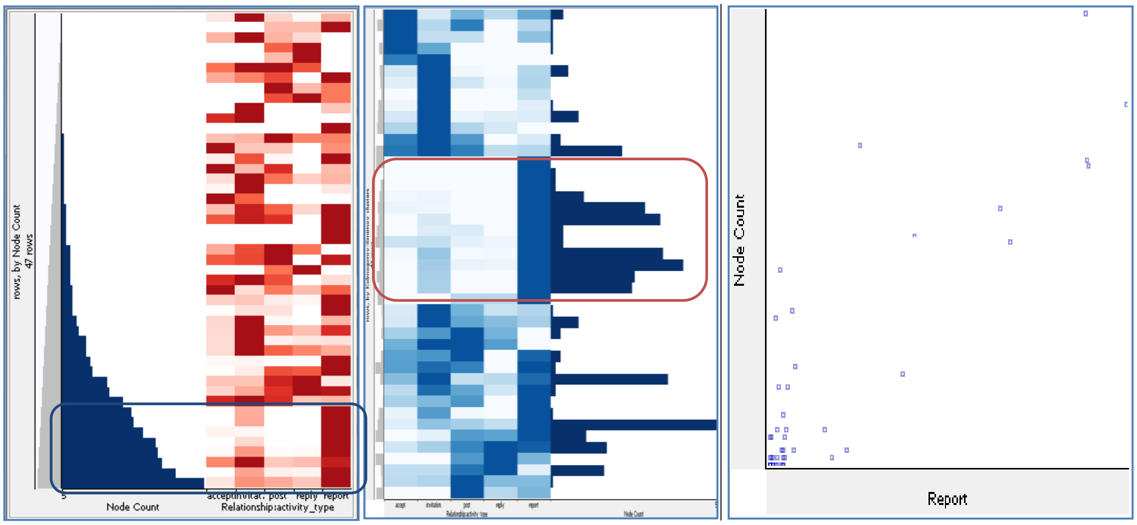
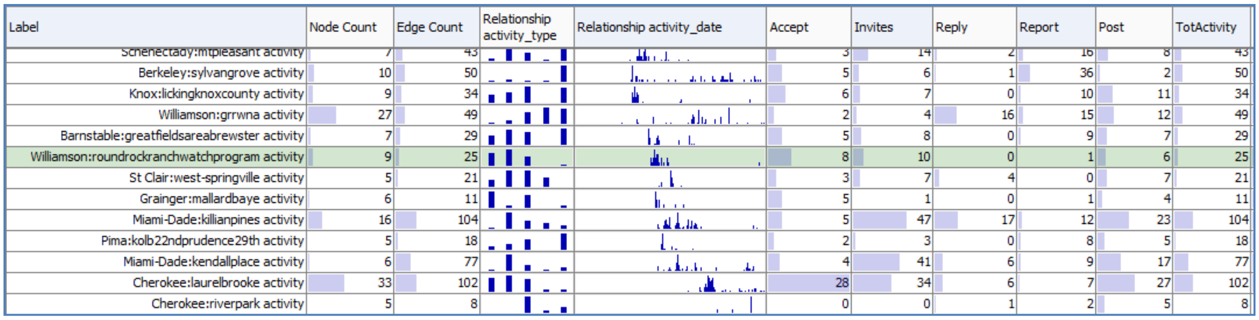


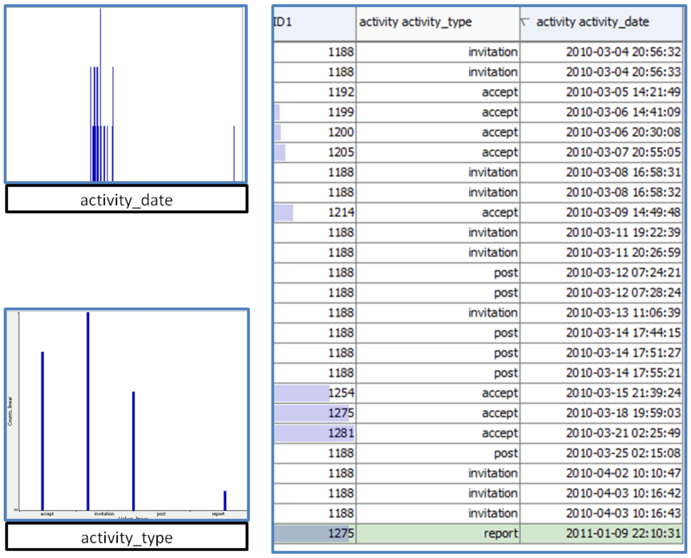
Figure 9: Left) Communities with at least 5 members: Node count and activity_type overview side by side, rows sorted by node count. Larger communities are generating more reports and not generating significant number of posts and replies. Middle) Now the activity_type column is clustered and the member count (nodes) for each community is shown side by side. Readers can observe groups of communities with similar activity pattern, some having more invitations, some having large proportion of reports, and others having a mixture of various activities. In both the overviews communities with more reports also tend to have more members. Dark shades of color in the heatmap indicate higher percentage of an activity type. Right) Now to verify this further I have drawn a scatter-plot in ManyNets showing node vs report counts in the communities.

Growth Pattern of Communities: Using the activity_date distribution column analysts can observe the rise and fall in activity over time, this column shows the total count of activity per day for each community, starting from July 2009 to February 2011. It shows different patterns in the activity. In some communities there was sudden spike in activity which did not persist throughout the time period. And in some communities the activity pattern is persistent. Then again some communities started with high activity but gradually their activity diminished. Also this view can help with outlier detection. For example, in the activity_date distribution column the network table Williamson:roundrockranchwatchprogram community has a spike of activity and then there is no activity for a long period and finally a small spike. To analyze what happened to this community, I opened the edge table for this community that shows the date and type for each activity (Figure 10) . And now I can see since the community started user 1188 has invited other users and posted in the community, while other users accepted invitation from 1188, and it all happened in March and April 2010, then nothing happened until January 2011, when user 1275 made a report which happens to be the only report in this community.



Label	Node Count	Edge Count	Relationship activity_type	Relationship activity_date	Accept	Invites	Reply	Report	Post	TotActivity
Schenectady:mtpleasant activity	7	43			3	14	2	16	8	43
Berkeley:sylvangrove activity	10	50			5	6	1	36	2	50
Knox:lickingknoxcounty activity	9	34			6	7	0	10	11	34
Williamson:grrwna activity	27	49			2	4	16	15	12	49
Barnstable:greatfieldsareabrewster activity	7	29			5	8	0	9	7	29
Williamson:roundrockranchwatchprogram activity	9	25			8	10	0	1	6	25
St Clair:west-springville activity	5	21			3	7	4	0	7	21
Grainger:mallardbaye activity	6	11			5	1	0	1	4	11
Miami-Dade:killianpines activity	16	104			5	47	17	12	23	104
Pima:kolb22ndprudence29th activity	5	18			2	3	0	8	5	18
Miami-Dade:kendallplace activity	6	77			4	41	6	9	17	77
Cherokee:laurelbrooke activity	33	102			28	34	6	7	27	102
Cherokee:riverpark activity	5	8			0	0	1	2	5	8

Figure 10: Activty_date column showing the temporal pattern of the activities in each community. Williamson:roundrockranchwatchprogram community is chosen for further analysis.



ID1	activity activity_type	activity activity_date
1188	invitation	2010-03-04 20:56:32
1188	invitation	2010-03-04 20:56:33
1192	accept	2010-03-05 14:21:49
1199	accept	2010-03-06 14:41:09
1200	accept	2010-03-06 20:30:08
1205	accept	2010-03-07 20:55:05
1188	invitation	2010-03-08 16:58:31
1188	invitation	2010-03-08 16:58:32
1214	accept	2010-03-09 14:49:48
1188	invitation	2010-03-11 19:22:39
1188	invitation	2010-03-11 20:26:59
1188	post	2010-03-12 07:24:21
1188	post	2010-03-12 07:28:24
1188	invitation	2010-03-13 11:06:39
1188	post	2010-03-14 17:44:15
1188	post	2010-03-14 17:51:27
1188	post	2010-03-14 17:55:21
1254	accept	2010-03-15 21:39:24
1275	accept	2010-03-18 19:59:03
1281	accept	2010-03-21 02:25:49
1188	post	2010-03-25 02:15:08
1188	invitation	2010-04-02 10:10:47
1188	invitation	2010-04-03 10:16:42
1188	invitation	2010-04-03 10:16:43
1275	report	2011-01-09 22:10:31

Figure 11: Williamson:roundrockranchwatchprogram community. Left) histogram at top shows activity_date and the bottom one shows the distribution of activity_type, invitations being the most occurred activity in this network. Right) Activity table showing the details with the member id, activity type and date.

While the distribution inside the table was useful in this particular case, to see all the communities in this column, users need the overview. So after generating the heatmap column overview, I first sorted the overview [Figure 12 a] according to the first activity that happened in each community. This way the users can see when the communities started to get active and how the activities grew or diminished over time. The rows in the top of the overview are all different communities from

Jefferson County, they started earlier than most other communities and continued to be active whereas, some communities are not active anymore. To see which communities became stale I have sorted the overview by the last date of activity [Figure 12 b]. By selecting the rows from this overview highlights the communities in the main table to show more information about them.

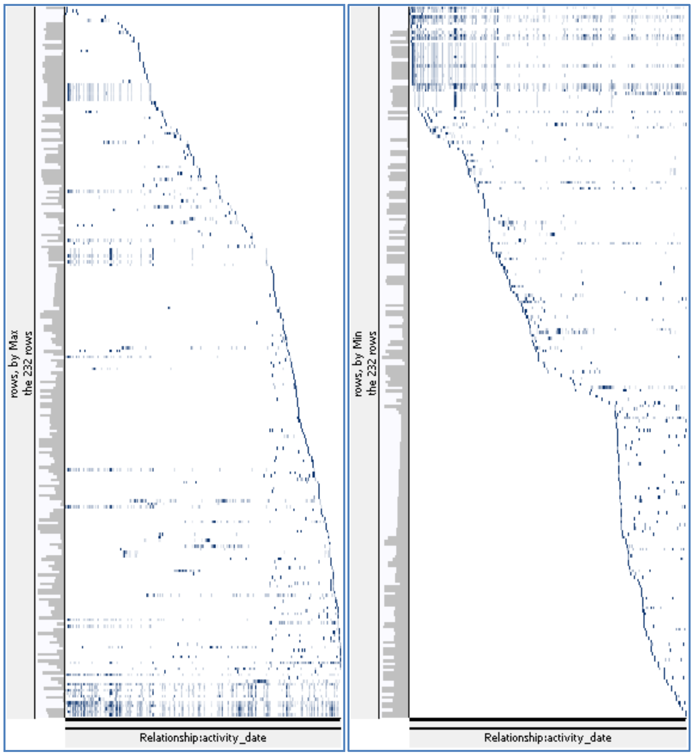


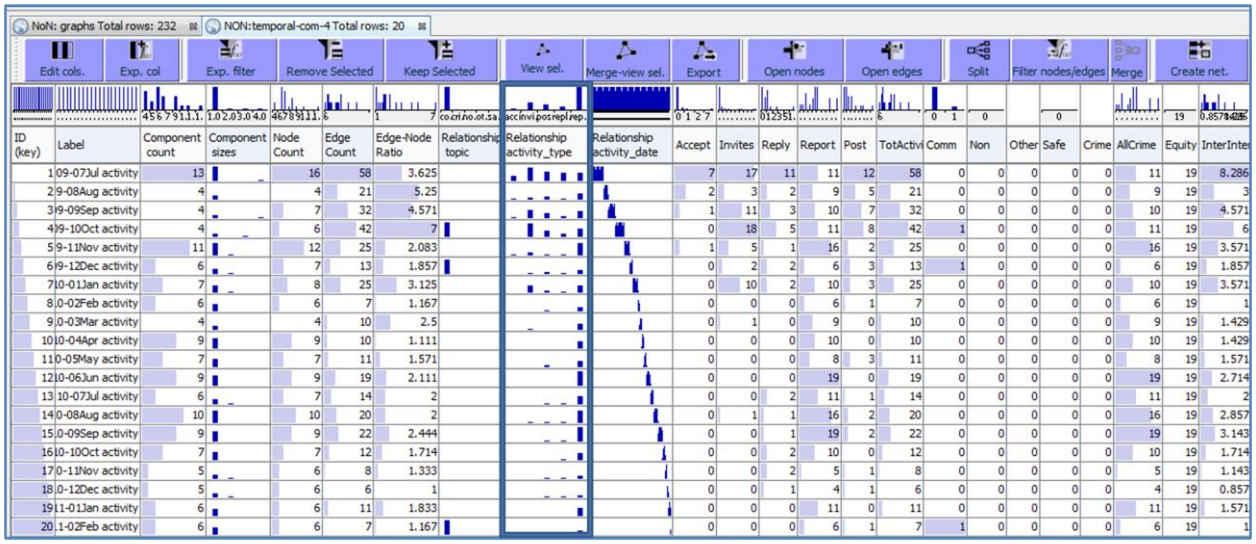
Figure 12: Heatmap overview of activity_date distribution column. a) Community activity sorted by the first activity. b) Communities sorted by their last activity.

This analysis is important to categorize communities; our analysis can categorize two types of communities according to the growth pattern:

1) Baby community: Communities that started 6 months ago or later.
2) Mature community: Communities those are older than 6 months.

For mature communities it is important to see if the members are communicating with each other or if they are posting reports. For new communities it is important to see if the members are trying to invite new members or not. Using the overview I can detect and select the mature and baby communities and then filter them in the main table to analyze separately.

Community Activity over Time: To analyze the growth of a particular community, it can be split into temporal snapshots. Each split containing activity of a month. After sampling the communities, Watch Jefferson County appears to be the most active one, and then I split it to observe its activity over time. In Figure 13, each row represents the network for a month, sorted by time from July 2009 to February 2011. Users can see that initially there were different types of activities but gradually the proportion of reports grew larger while no more invitations and acceptance occurred lately.

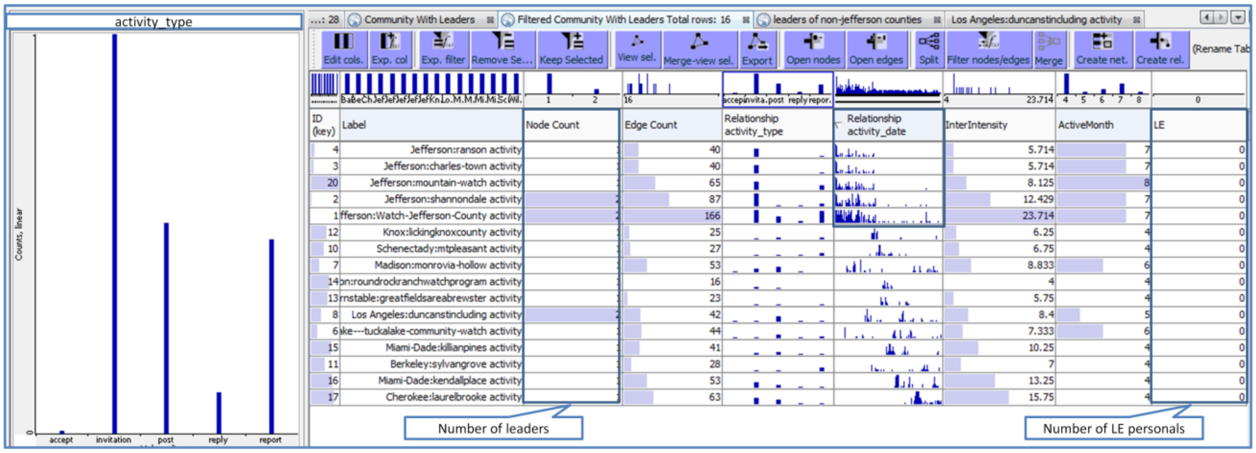


ID (key)	Label	Component count	Component sizes	Node Count	Edge Count	Edge-Node Ratio	Relationship topic	Relationship activity_type	Relationship activity_date	Accept	Invites	Reply	Report	Post	TotActivi	Comm	Non	Other	Safe	Crime	AllCrime	Equity	InterInte
1	09-07Jul activity	13		16	58	3.625				7	17	11	11	12	58	0	0	0	0	0	11	19	8.286
2	9-08Aug activity	4		4	21	5.25				2	3	2	9	5	21	0	0	0	0	0	9	19	3
3	9-09Sep activity	4		7	32	4.571				1	11	3	10	7	32	0	0	0	0	0	10	19	4.571
4	09-10Oct activity	4		6	42	7				0	18	5	11	8	42	1	0	0	0	0	11	19	6
5	9-11Nov activity	11		12	25	2.083				1	5	1	16	2	25	0	0	0	0	0	16	19	3.571
6	9-12Dec activity	6		7	13	1.857				0	2	2	6	3	13	1	0	0	0	0	6	19	1.857
7	10-01Jan activity	7		8	25	3.125				0	10	2	10	3	25	0	0	0	0	0	10	19	3.571
8	0-02Feb activity	6		6	7	1.167				0	0	0	6	1	7	0	0	0	0	0	6	19	1
9	0-03Mar activity	4		4	10	2.5				0	1	0	9	0	10	0	0	0	0	0	9	19	1.429
10	10-04Apr activity	9		9	10	1.111				0	0	0	10	0	10	0	0	0	0	0	10	19	1.429
11	0-05May activity	7		7	11	1.571				0	0	0	8	3	11	0	0	0	0	0	8	19	1.571
12	10-06Jun activity	9		9	19	2.111				0	0	0	19	0	19	0	0	0	0	0	19	19	2.714
13	10-07Jul activity	6		7	14	2				0	0	2	11	1	14	0	0	0	0	0	11	19	2
14	0-08Aug activity	10		10	20	2				0	1	1	16	2	20	0	0	0	0	0	16	19	2.857
15	0-09Sep activity	9		9	22	2.444				0	0	1	19	2	22	0	0	0	0	0	19	19	3.143
16	10-10Oct activity	7		7	12	1.714				0	0	2	10	0	12	0	0	0	0	0	10	19	1.714
17	0-11Nov activity	5		6	8	1.333				0	0	2	5	1	8	0	0	0	0	0	5	19	1.143
18	0-12Dec activity	5		6	6	1				0	0	1	4	1	6	0	0	0	0	0	4	19	0.857
19	11-01Jan activity	6		6	11	1.833				0	0	0	11	0	11	0	0	0	0	0	11	19	1.571
20	1-02Feb activity	6		6	7	1.167				0	0	0	6	1	7	1	0	0	0	0	6	19	1

Figure 13: Temporal split for WatchJeffersonCounty community. Activity type pattern changed over time.

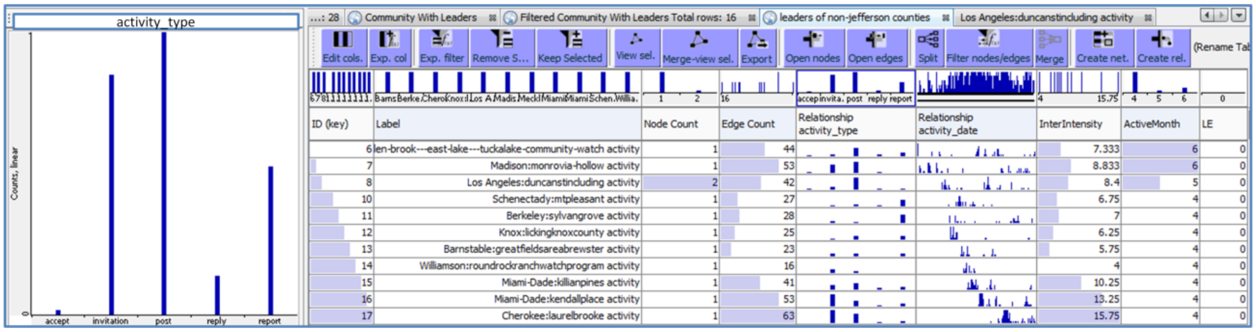
Leadership Analysis: Adding the significance metric made it easier to identify the leaders for each community. Now to analyze only the leaders I used the expression filtering mechanism provided by ManyNets which also lets me filter out nodes having particular property. So I have filtered out the members whose significance value is below zero.

Now the remaining communities are only those who have at least one leader and the community table contains the networks generated by the leaders and their activity. Now users can see only 16 communities have at least one leader and exactly 3 of them has 2 leaders, others having just one; and none of them has any member from law and enforcement. So it's not the law enforcement people who are leading the communities. From the histogram overview of activity type, users can also see that among the activities performed only by the leaders, invitations are most prevalent. That indicates leaders are trying to recruit new members in the community.



ID (key)	Label	Node Count	Edge Count	Relationship activity_type	Relationship activity_date	InterIntensity	ActiveMonth	LE
4	Jefferson:ranson activity	1	40			5.714	7	0
3	Jefferson:charles-town activity	1	40			5.714	7	0
20	Jefferson:mountain-watch activity	1	65			8.125	8	0
2	Jefferson:shannondale activity	2	87			12.429	7	0
1	fferson:Watch-Jefferson-County activity	2	166			23.714	7	0
12	Knox:lickingknoxcounty activity	1	25			6.25	4	0
10	Schenectady:mtpleasant activity	1	27			6.75	4	0
7	Madison:monrovia-hollow activity	1	53			8.833	6	0
14	on:roundrockranchwatchprogram activity	1	16			4	4	0
13	rnstable:greatfieldsareabrewster activity	1	23			5.75	4	0
8	Los Angeles:duncanstincluding activity	2	42			8.4	5	0
6	ake---tuckalake-community-watch activity	1	44			7.333	6	0
15	Miami-Dade:killianpines activity	1	41			10.25	4	0
11	Berkeley:sylvangrove activity	1	28			7	4	0
16	Miami-Dade:kendallplace activity	1	53			13.25	4	0
17	Cherokee:laurelbrooke activity	1	63			15.75	4	0

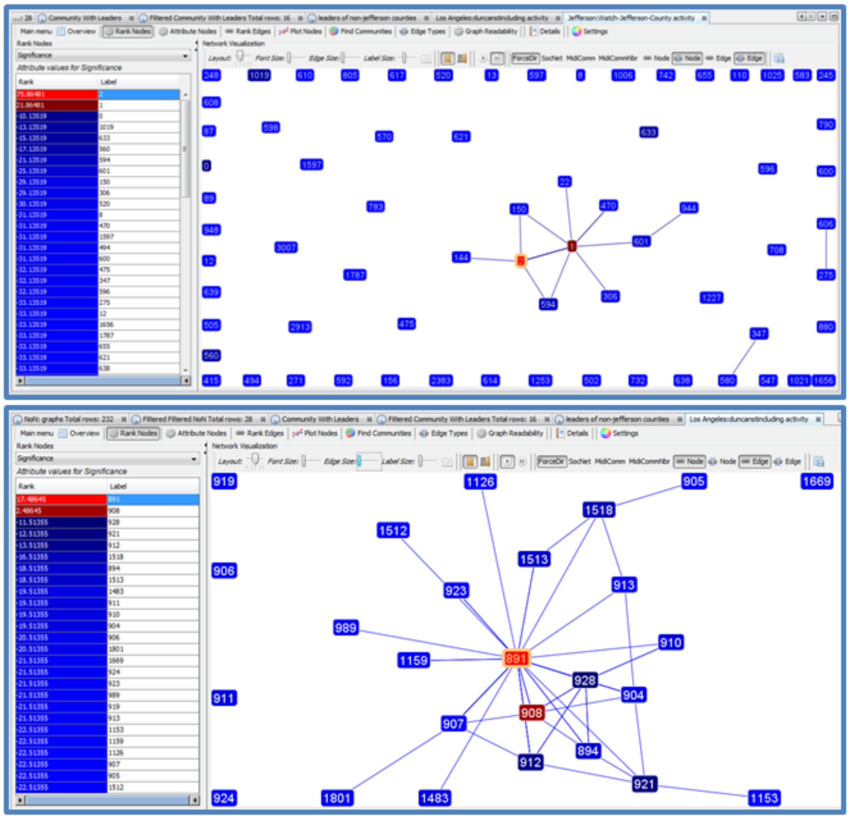
Figure 14: Leaders' activity network. 16 communities have leaders.



ID (key)	Label	Node Count	Edge Count	Relationship activity_type	Relationship activity_date	InterIntensity	ActiveMonth	LE
6	len-brook---east-lake---tuckalake-community-watch activity	1	44			7.333	6	0
7	Madison:monrovia-hollow activity	1	53			8.833	6	0
8	Los Angeles:duncanstincluding activity	2	42			8.4	5	0
10	Schenectady:mtpleasant activity	1	27			6.75	4	0
11	Berkeley:sylvangrove activity	1	28			7	4	0
12	Knox:lickingknoxcounty activity	1	25			6.25	4	0
13	Barnstable:greatfieldsareabrewster activity	1	23			5.75	4	0
14	Williamson:roundrockranchwatchprogram activity	1	16			4	4	0
15	Miami-Dade:killianpines activity	1	41			10.25	4	0
16	Miami-Dade:kendallplace activity	1	53			13.25	4	0
17	Cherokee:laurelbrooke activity	1	63			15.75	4	0

Figure 15: Leaders' activity network after excluding Jefferson County

In the activity_date column, users see the temporal patterns in the top 5 communities are very similar and they are all from Jefferson County. The leaders of communities are very active before and their activity decreased over time. To see if it has any effect on the overall leaders activity pattern, I have filtered out these communities and now users can see a different scenario, for the remaining communities the leaders are also involved in making posts and invitation is the second highest activity, followed by reports.



Rank	Label
75.86481	2
21.86481	1
-10.13519	0
-13.13519	1019
-15.13519	633
-17.13519	560
-21.13519	594
-25.13519	601
-29.13519	150
-29.13519	306
-30.13519	520
-31.13519	8
-31.13519	470
-31.13519	1597
-31.13519	494
-31.13519	600
-32.13519	475
-32.13519	347
-32.13519	596
-33.13519	275
-33.13519	12
-33.13519	1656
-33.13519	1787
-33.13519	655
-33.13519	621
-33.13519	638

Rank	Label
17.48645	891
2.48645	908
-11.51355	928
-12.51355	921
-13.51355	912
-16.51355	1518
-18.51355	894
-18.51355	1513
-19.51355	1483
-19.51355	911
-19.51355	910
-19.51355	904
-20.51355	906
-20.51355	1801
-21.51355	1669
-21.51355	924
-21.51355	923
-21.51355	989
-21.51355	919
-21.51355	913
-22.51355	1153
-22.51355	1159
-22.51355	1126
-22.51355	907
-22.51355	905
-22.51355	1512

Figure 76: node-link diagram for community activity: top) WatchJeffersonCounty . bottom) LosAngelesDuncuns.

After observing this I decided to see the network structure and for this, I selected one community from Jefferson county (WatchJefferson) and another from the remaining ones (Los Angeles). After opening the networks in node-link view, and then ranking the nodes by significance (red indicting the node with maximum significance, i.e., the leader), users can compare and contrast the leaders' role in these communities more clearly. In WatchJeffersonCounty, many nodes are disconnected, that is they are creating reports rather than having conversation. In contrast, in LosAngeles members are having more conversation (post and replies) and the leader member is also well-connected in the conversation, meaning s/he is making posts and replying to others' posts.

CONCLUSION

The network table, Node table and Edge table show different levels of aggregations and information, the NetBeans platform allows us to have new windows and tabs on demand, retaining the original table while creating new tabs for filtered rows; no other network visualization tool has this capability. In NodeXL[20] the filtering is done in the same table therefore to create new filtered subset of the nodes and edges users need to remove the filter from the original table and then perform new filtering operation. Moreover users can export the data from ManyNets in tabular format and use it in spreadsheet applications like Excel for regression analysis. Having the capability to generate both statistical and visual insights integrated in the same tool along with its filtering features provided the leverage of rapid reiteration within one tool without going back and forth among several tools.

Also I received feedback from Kent Norman, PJ Rey, Alan Neustadtl and Catherine Plaisant while working with this dataset. I got several improvement suggestions and have implemented some of them already while others will be implemented gradually. One of the suggestions is to have the distribution columns to be expanded on demand, so now the activity_type column shows the distribution of activities as histogram inside table cell; the expansion option will generate new columns for each type of activity per user request.

ManyNets supports different types of datasets as input and also lets the users create new relationships, but due to this feature, users also need to upload a schema, create node and edge lists for each network in a separate file, and need to have an xml description of the files for each network. As it does not use any database at the backend, the in memory calculation creates undesirable latency and memory consumption; for professional level use, a back end database server will be more useful. The python expression column is very useful for expert users. The analysis procedure is helpful but the user interface has a lot of room for improvement.

Using ManyNets and its column overview I analyzed NON dataset; emphasizing on comparing community and their temporal evolution along with leadership detection. The next step is to help PJ Rey to run regression analysis for this dataset to see the correlation of variables associated with the community growth. For this the metrics value calculated in ManyNets will be exported for the regression. I used the data obtained from 2005 to February 2011 for this report. As the community portal is still active I will collect more data (upto October 2011) and will analyze that to see if the patterns changed over this time.

Acknowledgement: Special thanks to Jae-wook Ahn for helping with the data formatting and database management. I would also like to thanks Catherine Plaisant and PJ Rey for their valuable suggestions.

REFERENCE

[1] Tableau Software. http://www.tableausoftware.com/. [Online; accessed 21-Mar-2010].

[2] TIBCO Spotfire. http://spotfire.tibco.com/. [Online; accessed 21-Mar-2010].

[3] Z. Bar-Joseph, D. K. Gifford, and T. Jaakkola. Fast optimal leaf ordering for hierarchical clustering. pages 22–29, (2001).

[4] R. Gentleman, V. Carey, D. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, K. Hornik, T. Hothorn, W. Huber, S. Iacus, R. Irizarry, F. Leisch, C. Li, M. Maechler, A. Rossini, G. Sawitzki, C. Smith, G. Smyth,L. Tierney, J. Yang, and J. Zhang. Bioconductor: open software development for computational biology and bioinformatics. Genome Biology, 5(10):R80,(2004).

[5] A. Hanson, "Nation of neighbors," http://nationofneighbors.com.

[6] GroupLens Research Project. Movielens 100K Dataset. http://www.grouplens.org/node/73. [Online; accessed 21-Mar-2010].

[7] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In Proceedings of the 1st Conference on Visualization'90, 378. IEEE Computer Society Press, (1990).

[8] R. Kincaid and H. Lam. Line graph explorer: scalable display of line graphs using focus+context. In AVI '06: Proceedings of the working conference on Advanced visual interfaces, 404–411, New York, NY, USA, (2006). ACM.

[9] A. Kobsa. An empirical comparison of three commercial information visualization systems. Proceedings of InfoVis, IEEE Symposium on Information Visualization. 123-130,(2001).

[10] R. Rao and S. K. Card. The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In CHI '94: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM Press.318–322, New York, NY, USA, (1994).

[11] J. Seo and B. Shneiderman. Knowledge discovery in high-dimensional data: Case studies and a user survey for the rank-by feature framework. IEEE Transactions on Visualization and Computer Graphics, 12(3):311–322, (May/June 2006).

[12] M. Spenke. Visualization and interactive analysis of blood parameters with infozoom. Artificial Intelligence in Medicine, 22(2):159–172, (2001).

[13] Doreian, P., Kapuscinski, R., Krackhardt, D., Szczypula, J.: A brief history of balance through time. The Journal of Mathematical Sociology 21(1), 113–131 (1996).

[14] Durant, K., McCray, A., Safran, C.: Modeling the temporal evolution of an online cancer forum. In: 1st ACM International Health Informatics Symposium (IHI 2010). 356–365 (November 2010).

[15] Freire, M., Plaisant, C., Shneiderman, B., Golbeck, J.: Manynets: an interface for multiple network analysis and visualization. In: Proceeding of the 28th annual SIGCHI Conference on Human Factors in Computing Systems. 213–222 (2010).

[16] A. Perer and B. Shneiderman: Integrating Statistics and Visualization: Case Studies of Gaining Clarity during Exploratory Data Analysis. SIGCHI Conference on Human Factors in Computing Systems (CHI 2008) .

[17] Gloor, P., Laubacher, R., Zhao, Y., Dynes, S.: Temporal visualization and analysis of social networks. In: NAACSOS Conference, June. 27–29 (2004).

[18] Gloor, P., Zhao, Y.: Analyzing actors and their discussion topics by semantic social network analysis. In: Information Visualization, 2006. IV 2006. 130–135. IEEE (2006).

[19] http://tangow.ii.uam.es/mn/wiki/UserManual

[20] Hansen, D.L., Shneiderman, B., Smith, M.A.: Analyzing Social Media Networks with NodeXL: Insights from a Connected World. Morgan Kaufmann (2010).

[21] Moody, J., McFarland, D., Bender-deMoll, S.: Dynamic network visualization. American Journal of Sociology 110(4), 1206–41 (2005).

[22] Perer, A., Shneiderman, B.: Integrating statistics and visualization: case studies of gaining clarity during exploratory data analysis. In: Proceeding of the 26th annual SIGCHI conference on Human factors in computing systems. 265–274 (2008).

[23] Powell, W., White, D., Koput, K., Owen-Smith, J.: Network Dynamics and Field Evolution: The Growth of Interorganizational Collaboration in the Life Sciences. American Journal of Sociology 110(4), 1132–1205 (2005).

[24] Trier, M.: Towards dynamic visualization for understanding evolution of digital communication networks. Information Systems Research 19(3), 335–350 (2008).

[25] prefuse. http://prefuse.org/

[26] Sharara H., Sopan A., Namata G., Getoor L., Singh L.: G-PARE: A Visual Analytic Tool for Comparative Analysis of Uncertain Graphs . IEEE Conference on Visual Analytics Science and Technology (2011) .

[27] Ahn, J., Taieb-Maimon, M., Sopan, A., Plaisant, C., Shneiderman, B.: Temporal Visualization of Social Network Dynamics: Prototypes for Nation of Neighbors. Proc. Of Social Computing, Behavioral-Cultural Modeling and Prediction Conference. 309-316,(2011).

[28] Sopan, A., Freire, M., Taieb-Maimon, M., Golbeck, J., Shneiderman, B., Shneiderman, B.: Exploring Distributions: Design and Evaluation.HCIL tech report(2010).

[29] Peterson, E.:Time spring layout for visualization of dynamic social networks. Network Science Workshop (NSW), (2011).

[30] ManyNets.http://www.cs.umd.edu/hcil/manynets/

[31] ManyNets video demo.http://www.cs.umd.edu/hcil/manynets/videos/ManyNets-09-15-09_flash/second-video.mp4

[32] Preece, J. Sociability and usability in online communities: Determining and measuring success, Behaviour & Information Technology, 20, 5, 347(2001).

[33] Butler, B.S. Membership size, communication activity, and sustainability: A resource-based model of online social structures, Information Systems Research, 12, 4, 346-362 (2001).