Analysis of Communities Evolution in Dynamic Social Networks

Nikolai Nefedov

Abstract. In this paper we present a framework to study evolution of communities in dynamic networks. A dynamic network is represented by a sequence of static graphs named as network snapshots. We introduce a distance measure between static graphs to study similarity among network snapshots and to detect outlier events. To find a detailed structure within each network snapshot we used a modularity maximization algorithm based on a fast greedy search extended with a random walk approach. Community detection often results in a different number of communities in different network snapshots. To make communities evolution studies feasible we propose a greedy method to match clustering labels assigned to different networks. The suggested framework is applied for analysis of dynamic networks built from real-world mobile datasets.

1 Introduction

The growing spread of smart phones equipped with various sensors makes it possible to record rich-content user data and complement it with on-line processing. Mobile data processing could help people to enrich their social interactions and improve environmental and personal health awareness. At the same time, mobile sensing data could help service providers to understand better human behavior and its dynamics, identify complex patterns of users' mobility, and to develop various service-centric and user-centric mobile applications and services on-demand. One of the first steps in analysis of rich-content mobile datasets is to find an underlying structure of users' interactions and its dynamics by clustering data according to some similarity measures. In cases when data are given in the relational format (causality or dependency relations), e.g., as a network consisting of N nodes and E edges representing some relations among the nodes, then this task may be formulated as a problem of finding

Nikolai Nefedov
ISI Lab., Swiss Federal Institute of Technology, Zurich (ETHZ)
e-mail: nefedov@isi.ee.ethz.ch

G. Ghoshal et al. (Eds.): *Complex Networks IV*, SCI 476, pp. 39–46.
DOI: 10.1007/978-3-642-36844-8_4

communities, i.e., groups of nodes which are interconnected more densely among themselves than with the rest of the network.

The growing interest to the problem of community detection was triggered by the introduction of a new clustering measure called modularity [1]. The direct modularity maximization is known as a NP-hard problem and currently a number of sub-optimal algorithms are proposed, e.g., see [2] and references within. However, most of these methods address static networks partitioning into disjoint communities. On the other hand, in practice communities are dynamic and often overlapping structures. It is especially visible in social networks, where interactions among people and their affiliations to different groups are changing in time.

In this paper we present a framework to study evolution of communities in dynamic networks. A dynamic network is represented by a sequence of static graphs named as network snapshots [3]. We introduce a distance measure between static graphs to study similarity among network snapshots and to detect outlier events. To find a detailed structure within each network snapshot we used a modularity maximization algorithm based on a fast greedy search [4, 5] extended with a random walk approach [6, 7]. Community detection may results in a different number of communities in each network snapshot and in a different labeling of communities within snapshots. To make communities evolution visible we propose a greedy method to match clustering labels assigned to different network snapshots. The paper is organized as follows. In Section 2 we describe a distance measure between networks based on graph Laplacian spectra. A greedy algorithm to match partitions is outlined in Section 3. Analysis of real-world mobile datasets [8] briefly presented in Section 4, followed by conclusions in Section 5.

2 Distance Measure between Networks

To quantify structural properties of dynamic networks a variety of measures has been suggested. For example, in [9] a measure based on Katz-centrality is proposed to analyze time-dependent networks. However, this measure assumes a connected network, which is not always observed in dynamic social or biological networks with a set of disjoint subgraphs. On the other hand, substructure-based measures (e.g, edit-distance, a maximal common subgraph) do not take into account a global structure of a graph. Furthermore, usually only a part of users (nodes) appear in a network snapshot, a total set of nodes is obtained only after the aggregation of all snapshots.

In this paper we use graph spectral methods [10, 11] to characterize global graph structures (e.g., a graph connectivity, disjoint subgraphs) and compare network snapshots defined on a common set of nodes. Graph Laplacian is widely used to describe network structure, but its discrete nature complicates networks comparison. To compare network snapshots aggregated over different time periods we used dynamical systems approach similar to [12, 13].

Let us consider a network of N identical particles (nodes) connected by elastic strings according to an adjacency matrix $\mathbf{A}$ and described by motion equations

$$\ddot{x}_i + \sum_{j=1}^{N-1} A_{ij}(x_i - x_j) = 0\,, \tag{1}$$

where x_i is the coordinate of the i-th particle. Vibrational frequencies ω_a of this network are defined by eigenvalues $\gamma_a = -\omega_a^2$ of Laplacian $\mathbf{L}_A$ of the matrix $\mathbf{A}$. Laplacian spectrum of a graph is often called a vibrational spectrum [10]. In the following we measure a similarity between two graphs using Laplacian spectra. In particular, we present a spectral density $\rho(\omega)$ of a graph G as a sum of narrow Lorentz distributions [14]

$$\rho(\omega) = K \sum_{a=1}^{N-1} \frac{\gamma}{(\omega - \omega_a)^2 + \gamma^2}\,, \tag{2}$$

where γ is the width of the Lorentz distributions, K is a normalization coefficient such that $\int \rho(\omega)d\omega = 1$. Using spectral densities (2), a distance $d(G_k, G_m)$ between two graphs G_k and G_m may be defined using the mean square error

$$d_e(G_k, G_m) = \int_0^\infty [\rho_k(\omega) - \rho_m(\omega)]^2 d\omega\,, \tag{3}$$

or as the inner product of densities

$$d_p(G_k, G_m) = \sum_i \rho_k(\omega_i) \cdot \rho_m(\omega_i). \tag{4}$$

In this paper we use only (4) for networks comparison.

3 Partitions Matching Algorithm

In general, subgraphs matching is a NP-hard problem. In the following we used a greedy matching strategy to find sub-optimal solutions. To match partition labels over all network snapshots we process iteratively two graphs at a time. A simplified description of one iteration is outlined below.

Greedy algorithm to match partition labels in two graphs

Input: partition matrix $P(N,2)$, where $P(:,1)$ is formed by partition labels of a reference graph, $P(:,2)$ consists of community labels of a graph to be matched; labels corresponding to unconnected nodes in $P(:,2)$ are set to zero.

Initialization:
- find indexes of nodes for each of the communities in $P(:,1)$ and $P(:,2)$;
- mark all communities in $P(:,2)$ as unprocessed;

Repeat until all communities are marked as processed in $P(:,2)$:
- select a set of unprocessed communities in $P(:,2)$;

```
- find a community $c_2(k)$ with the largest number of same labels $l_2^{(m)}$ in $P(:,2)$;
- set $l_2^{(swap)} = l_1(k)$, where $l_1(k)$ corresponds to $k$-th community label in $P(:,1)$;
- swap labels $l_2^{(m)}$ and $l_2^{(swap)}$ in $P(:,2)$;
- mark community $c_2(k)$ in $P(:,2)$ as processed.
Stop when a maximum number of iterations is reached.
```

We tested the algorithm using synthesized networks. The greedy matching finds the optimal solution in 80% cases, in other cases solutions are close to the optimal. The complexity of the algorithm is mainly determined by a finite number of selection operations (sorting and swapping), which is in average $\mathscr{O}(N \log N)$.

4 Analysis of Real World Datasets

To analyze mobile users behavior and underlying social structure Nokia/Lausanne organized a mobile data collection campaign (MDCC) at EPFL university campus [15]. Rich-content datasets (including data from mobile sensors, call-logs, users proximity, their locations and etc) are collected from about 200 participants during June/2009-June/2011 [15].

Below we briefly outline applications of the proposed framework for analysis of dynamic social affinity graphs constructed from MDCC voice-call logs. Fig.1 shows network snapshots constructed by aggregating voice-call interactions among MDCC participants during different months. First, we analyzed a similarity among network

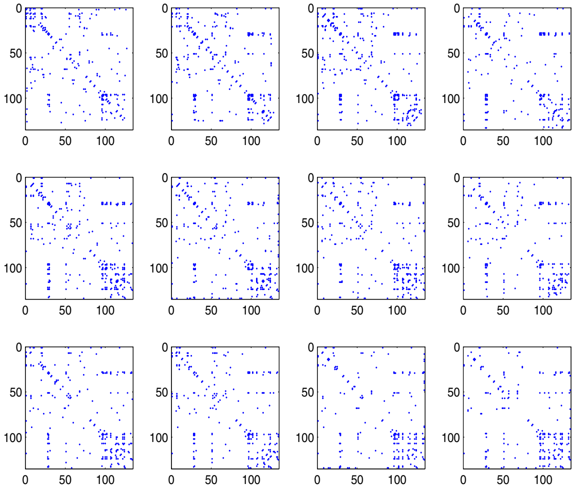

Fig. 1 Dynamics of voice-call activities among MDCC participants during Jan-Dec/2010: adjacency matrices are aggregated over one month period

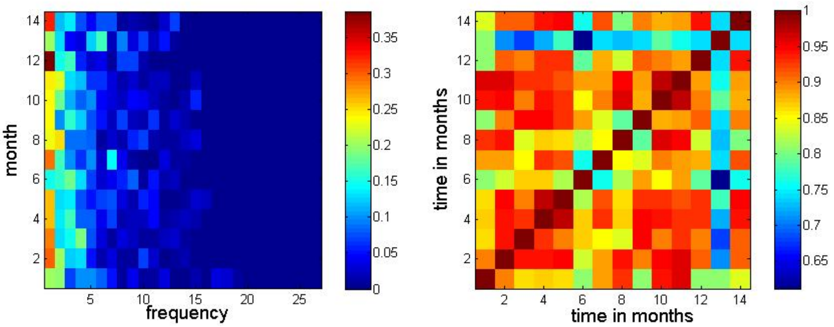

Fig. 2 Normalized vibration spectra (2) of monthly-aggregated network snapshots during Jan/2010-Feb/2011 (left part). Distance d_p among different network snapshots (right part), it clearly shows an outlier at ΔT_{13}.

snapshots using the proposed distance measures described above. As an illustration, vibration spectra (2) and distance (4) among monthly-aggregated network snapshots are depicted at Fig.2, left and right parts, respectively. In particular, at the right part of Fig.2 one can see a high similarity ($d_p = 0.9 \ldots 0.95$) in social interactions for periods $\Delta T_2, \ldots, \Delta T_5$ (Feb-May/2010) and $\Delta T_9, \ldots, \Delta T_{12}$ (Sept-Dec/2010). Since a significant part of MDCC participants are students, these similar behavior patterns most probably correspond to session periods at the EPFL university. Also, one can clearly see an abnormality in social interaction at ΔT_{13} (Jan/2011). Detailed inspection of the data revealed that during this period most of the participants were contacted by one of the organizers about the MDCC updated conditions. For the following analysis we removed network connections relevant to detected outlier events.

To find communities in each network snapshot we used the algorithm [4] extended with a random walk [6, 7]. Communities detected in the voice-call network for data aggregated over the whole data collection campaign are shown by different colors at Fig.3.

Next, we applied the community detection algorithm for network snapshots built from monthly data. Connectivity among participants and their numbers are different at each snapshot, it results in a different number of communities shown by different colors at each snapshot (Fig.4). Furthermore, even in cases when some nodes (users) happen to belong to the same community at different time periods, their community labels assigned by community detection at different network snapshots may not necessarily coincide. As an illustration, color-coded community labels in different network snapshots corresponding to different months are shown at Fig.5, left. Dark blue color here (marked by zero at color bar) indicates no-calls intervals for a user within the participants set.

Hence, to analyze a community evolution we need to find a set of clustering labels at each network snapshot which gives the best match to clustering labels for a reference case. As a reference for MDCC datasets we used snapshots ag-

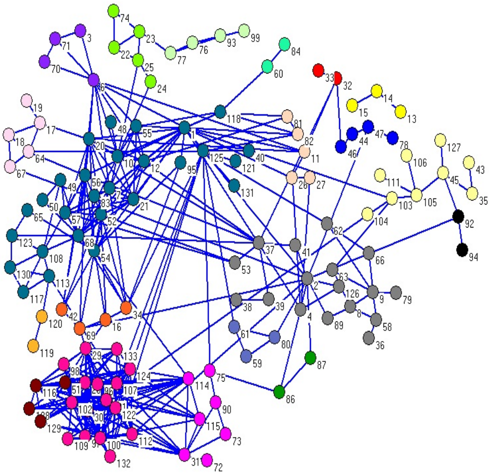

Fig. 3 Communities detection in voice-call network; data are aggregated over the whole MDCC period

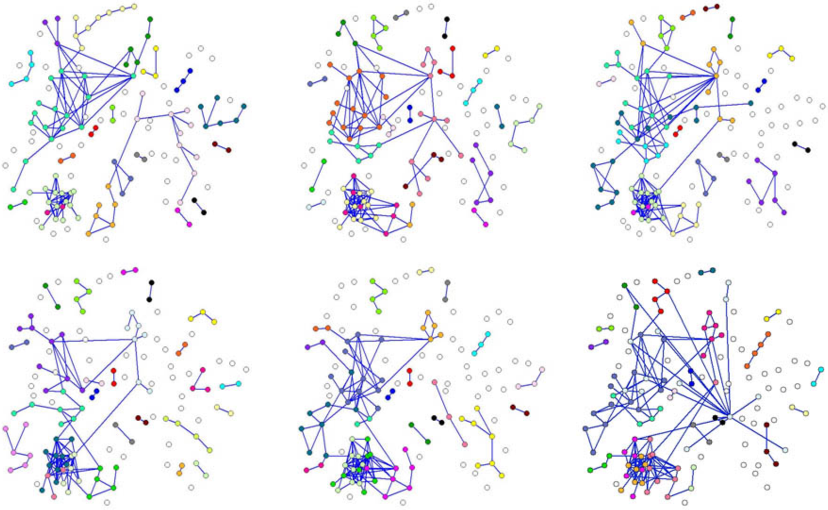

Fig. 4 Communities detection in voice-call networks, data are aggregated over one month period. Upper row: Jan-March/2010; lower row: Apr-June/2010.

gregated over the whole period (cf. Fig.3). Fig.5 (right part) depicts communities within monthly snapshots with community labels matched to the reference network. Columns on the left from color bars at Fig.5 present communities detected in the reference network. All participants are re-ordered according to community labels derived from the reference network. As one can see, after re-labeling the evolution of

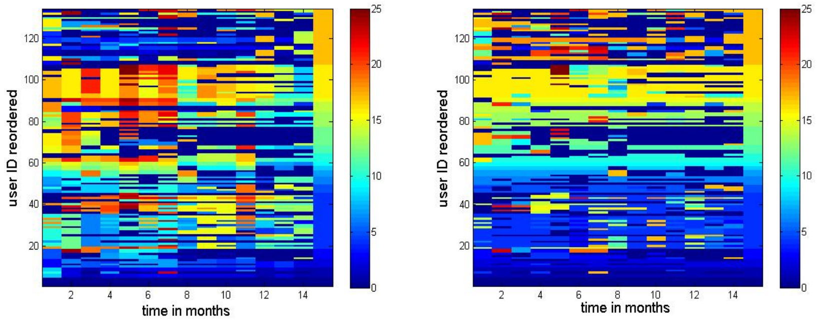

Fig. 5 Evolution of communities in MDCC voice-call social network during 14 months: color-coded community labels for 134 users in monthly-aggregated snapshots before (left part) and after (right part) re-labeling. Dark blue color (marked by zero at color bar) indicates no-calls intervals. The 15th column on both figures presents the assignment of community labels in the aggregated over the whole period network.

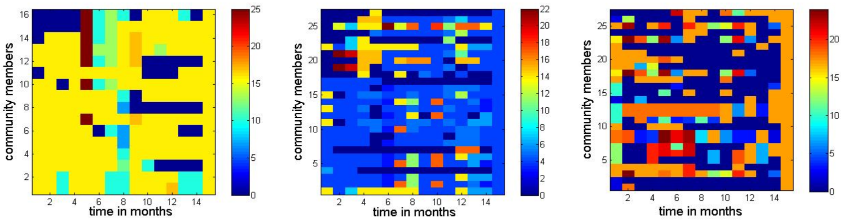

Fig. 6 Evolution of voice-call activity of users within their own communities in time. Examples of communities with dominating intra- and inter-community activities are shown on the left and on the right, respectively. Color bar represents community labels.

communities in time became clearly visible. Examples of communities with intra- and inter-communities interactions are depicted at Fig.6. One of the observations here is that communities detected in networks built from all aggregated data may be misleading. For example, the community shown at Fig.6 (right) actually is not observable in monthly network snapshots. In fact, it hardly may be called a homogenous community due to prevailing inter-community interactions, while users interactions within this community are sparse and not stable. It looks that this community actually is an artifact appeared due to data aggregation over a long period. On the other hand, the community at Fig.6 (left) reveals the stable structure at both, monthly and over the year, time scales.

5 Conclusions

In this paper we introduced a distance measure between network snapshots and applied it to study dynamics of communities in real-world mobile datasets. The proposed method allowed us to find outliers and clean the data. Community detection results in a different number of communities at different network snapshots. To match clustering labels at different snapshots we proposed a suboptimal greedy re-labeling method, verified it on synthesized networks and then applied it for real-world mobile data. The proposed method allowed us to remove artifacts in community detection due to data aggregation.

References

1. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E 69, 026113 (2004)
2. Fortunato, S.: Community detection in graphs. Physics Reports 486, 75–174 (2011)
3. Spiliopoulou, M.: Evolution in Social Networks: A Survey. In: Social Network Data Analytics, pp. 149–175. Springer Science+Business Media, LLC (2011)
4. Newman, M.E.J.: Fast algorithm for detecting community structure in networks. Physical Review E 69, 066133 (2004)
5. Blondel, V., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 1742-5468(10), P10008+12 (2008)
6. Lambiotte, R., Delvenne, J.C., Barahona, M.: Laplacian Dynamics and Multiscale Modular Structure in Networks, ArXiv:0812.1770v3 (2009)
7. Nefedov, N.: Multiple-Membership Communities Detection and its Applications for Mobile Networks. In: Applications of Digital Signal Processing, pp. 51–76. InTech (November 2011)
8. Nokia Mobile Data Challenge Campaign, `http://research.nokia.com/page/12000`
9. Grindrod, P., Higham, D.J., Parsons, M.C., Estrada, E.: Communicability across evolving networks. Phys. Rev. E 83, 046120 (2011)
10. Chung, F.R.K.: Spectral Graph Theory. CMBS Lectures Notes 92. AMS (1997)
11. Fay, D., et al.: Weighted Spectral Distributions: A Metric for structural Analysis of Networks. In: Statistical and Machine Learning Approaches for Network Analysis, pp. 153–190. John Wiley & Sons Inc., NY (2012)
12. Arenas, A., Diaz-Guilera, A., Kurths, J., Moreno, Y., Zhou, C.: Synchronization in complex networks. Physics Reports 469, 93–153 (2008)
13. Nefedov, N.: Applications of System Dynamics for Communities Detection in Complex Networks. In: IEEE Int. Conf. on Nonlinear Dynamics and Sync. (INDS 2011) (2011)
14. Ipsen, M., Mikhailov, A.: Evolutionary reconstruction of networks. Physical Review E 66, 046109 (2002)
15. Kiukkonen, N., Blom, J., Dousse, O., Gatica-Perez, D., Laurila, J.: Towards Rich Mobile Phone Datasets: Lausanne Data Collection Campaign. In: Proc. ACM Int. Conf. Pervasive Services, Berlin (2010)