

Community Change Detection in Dynamic Networks in Noisy Environment

Sadamori Koujaku Mineichi Kudo Ichigaku Takigawa Hideyuki Imai
Graduate School of Information Science and Technology,
Hokkaido University, Sapporo, Japan
{koujaku, mine}@main.ist.hokudai.ac.jp,
{takigawa, imai}@ist.hokudai.ac.jp

ABSTRACT

Detection of anomalous changes in social networks has been studied in various applications such as change detection of social interests and virus infections. Among several kinds of network changes, we concentrate on the structural changes of relatively small stationary communities. Such a change is important because it implies that some crucial changes have happened in a special group, such as dismissal of a board of directors. One difficulty is that we have to do this in a noisy environment. This paper, therefore, proposes an algorithm that finds stationary communities in a noisy environment. Experiments on two real networks showed the advantages of our proposed algorithm.

Categories and Subject Descriptors

H.2.8 [Data management]: Database Applications – Database

General Terms

Clustering, Data mining

Keywords

Social communication network, spatial-temporal analysis, event mining, community detection

1. INTRODUCTION

There has been considerable interest in the change analysis of social networks, such as communications among individuals through text messages [Newman et al. 2002], relations of opinions among bloggers [Gruhl et al. 2004, Agarwal et al. 2008], and collaboration among coauthored scientists [Newman 2006]. These networks change more or less their structure over time. Detection of their anomalous changes is of great interests across various applications such as transitions of social interests [Akoglu and Faloutsos 2008, Sun

et al. 2007], outbreaks of rumours and infections [Gruhl et al. 2004].

A dynamic network is typically represented by a time series of graphs. A change in a graph sequence falls into one of three main categories according to the structure level of interest: global structure, local structure and the community-level structure. Changes in global structure are detected by a statistic of the whole graph such as the *network diameter* [Leskovec et al. 2005]. This is useful to understand how the graph evolves as a whole. Changes in local structures are detected around a node of interest or a small subgraph of interest. Such a change detection is useful when we know in advance that some nodes or subgraphs have a high potential of change [Akoglu and Faloutsos 2008, Idé et al. 2007]. Changes in community-level structure are seen in between above two levels. This level focuses some densely connected groups of nodes, called *communities* [Newman 2006]. In social networks, one group of communities repeats merging/splitting or growing/shrinking, while another group of communities remains unchanged for a long time [Palla et al. 2007]. The former group are ad-hoc communities formed temporally with a short life. The latter group has a longer life and is often of key communities such as a board of directors [Priebe and Conroy 2006, Newman 2006] and a group of opinion leaders in blog networks [Agarwal et al. 2008]. Therefore, changes in such a stable community are one of important signs of crucial events in practice.

In this paper, we focus on community-level changes of communications among individuals through e-mails and phone-calls. In this network, a node is an individual and an edge represents the interaction between two individuals at one time. A group sharing a common interest/profit may have a strong connection among its members and exchange messages frequently, but the group is not so large in the scale of the whole network. In general such a network includes a noise, i.e., some messages may be wrongly sent, private contacts may be hidden, some rare case contacts or urgent inquiries from an extra group may occur. Therefore, one of the most challenging issues is to find small but stationary communities in a noisy environment.

To carry out change detection on stationary communities, we have to 1) establish a methodology to extract only stationary communities, along with an appropriate definition of a community, and 2) choose an appropriate statistic showing changes in each community, and 3) establish a way to detect changes according to the value of the statistic. In this paper, we mainly concentrate on the first issue and compare

several community models for stationary community extraction in noisy environment, and then apply them to change detection in a time series of graphs.

2. PRELIMINARIES

2.1 Notations

Let $G = (V, E)$ be an undirected graph where V is a set of nodes and $E \subseteq V \times V$ is a set of edges. We express the set of nodes by $V = \{1, 2, 3, \dots, |V|\}$ and assume no self loop. We express G by the adjacency matrix $A = (a_{ij})$ defined by

$$A = (a_{ij}), a_{ij} = \begin{cases} 1 & (e_{ij} \in E) \\ 0 & (\text{otherwise}) \end{cases}, \quad (1)$$

where e_{ij} is the edge between nodes i and j . For subsets $U, U_1, U_2 \subseteq V$, let us define the *within-set weight* $W(U)$ and the *between-set weight* $B(U_1, U_2)$ by

$$W(U) = \sum_{i,j \in U, i < j} a_{ij}, B(U_1, U_2) = \sum_{i \in U_1, j \in U_2} a_{ij}. \quad (2)$$

Suppose a dynamic system producing a random sequence of graphs $G^{(1)}, G^{(2)}, \dots, G^{(t)}, \dots$, where $G^{(t)} = (V, E^{(t)})$, $\forall t$. We assume the existence of a noise such that an edge in $E^{(t)}$ disappears with the probability ϵ ($0 < \epsilon < \frac{1}{2}$) and an absent edge $e' \notin E^{(t)}$ turns to an existing edge with the probability ϵ as well. This means that, by this noise, the probability p of the appearance of an edge in $E^{(t)}$ changes to p' as

$$p' = p(1 - \epsilon) + (1 - p)\epsilon = (1 - 2\epsilon)p + \epsilon. \quad (3)$$

Note that $p' > p$ for $p < 1/2$ and $p' < p$ for $p > 1/2$. That is, density becomes close to $1/2$ by this noise. We assume that edges occur at random according to the above probability, while V remains unchanged. Under this assumption, the expected graph $\bar{G} = (V, \bar{E})$ with the noise is an weighted clique with edge weight \bar{p}_{ij} between every node pair of i and j . The *finite sample-averaged* weight in time period $[1, t]$ is given by

$$\hat{p}'_{ij} := \frac{1}{t} \sum_{s=1}^t \mathbf{1}\{e_{ij} \in E^{(s)}\}, \quad (4)$$

where $\mathbf{1}$ is the indicator function. We denote $\hat{G}^{[1,t]}$ as the graph with edge weight \hat{p}'_{ij} .

2.2 Problem of Community Detection

Typically, the extraction of a family of communities $\mathcal{U} = \{U_1, U_2, \dots, U_K\}$, $\forall U_k \subseteq V$, is achieved by solving the following the maximization problem:

$$\max_{\mathcal{U}} Q(\mathcal{U}) = \max_{U_1, U_2, \dots, U_K} \sum_{i=1}^K q(U_i), \quad (5)$$

$$\text{s.t. } U_i \cap U_j = \emptyset, \forall i \neq j, \bigcup_{i=1}^K U_i = V, \quad (6)$$

where $q : 2^V \rightarrow \mathbb{R}$ is a given quality function for a community. The number K of communities is determined empirically or by a model selection criterion. This formulation is known as the *graph partitioning problem* [von Luxburg 2007]. In this problem, the key is how to choose $q(\cdot)$ appropriately.

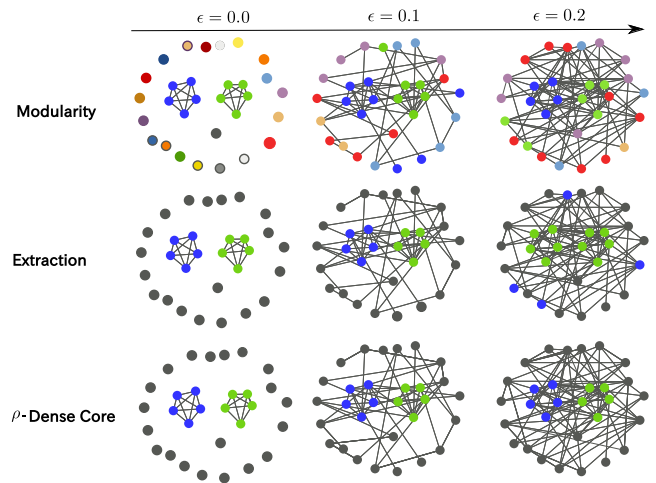


Figure 1: Communities found by Modularity, Extraction and ρ -Dense core ($\rho = 0.5$). The value of ϵ is the strength of noise. The color of nodes expresses the membership of the nodes, while a gray node does not belongs to any community.

Table 1: Summary of graph partitioning criteria

Aiming Community	Criterion (Who, When)	Quality function $q(U)$	Robustness
Aiming Community	Ratio Cut (Hagen, '92)	$\frac{B(U, \bar{U})}{ U }$	○
	EigenCluster (Sarkar, '96)	$\frac{W(U)}{ U }$	○
	Normalized Cut (Malik, '00)	$\frac{B(U, \bar{U})}{W(U) + B(U, \bar{U})}$	○
Balanced Community	Modularity (Newman, '04)	$W(U) - \frac{(W(U) + B(U, \bar{U}))^2}{2W(V)}$	×
	Graph Scan (*) (Wang, '08)	$\log \left(\frac{W(U)}{\mu(U)} \right)^{W(U)} + \log \left(\frac{W(V) - W(\bar{U})}{\mu(V) - \mu(\bar{U})} \right)^{(W(V) - W(\bar{U}))}$	×
Dense Community	Extraction (Zhao, '11)	$\frac{ \bar{U} }{ U } W(U) - B(U, \bar{U})$	×
	ρ-Dense Core (Koujaku, '14)	$W(U) - \frac{\rho}{2} U (U - 1)$	○

(*) $\mu(U)$ is the expected number of edges inside U defined as $\mu(U) = \frac{1}{2}|U|(|U| - 1)\eta(V)$.

3. CHANGE DETECTION OF COMMUNITIES

3.1 Community Models

Fig. 1 shows how stably three algorithms can find the communities in different levels of noises. Two main points of evaluation of algorithms are a) if they extract only stationary communities, and b) how robust they are against noise. From Fig. 1, we notice that 1) *Modularity* [Newman 2006] assigns forcibly every node to one of the communities (against our evaluation point a)), 2) *Extraction* [Zhao et al. 2011] is a little weak to the strong noise (against our evaluation point b)), and 3) *ρ -Dense core* [Koujaku et al.] succeeds to find only stationary communities in the noisy environment. We need such a robustness as the ρ -Dense core shows in noisy communication networks.

Table 1 summarizes the representative quality functions including the above three algorithms. Other than these algorithms, various models have been proposed such as *Ratio cut* [Hagen and Kahng 1992], *EigenCluster* [Sarkar and

Boyer 1996] and *Normalized cut* [Malik 2000], mainly in computer science. A group of them prefers to divide a graph into the subgraphs of almost equal sizes. Such a trial is effective in some applications such as an equal distribution of resources, for example, an assignment of tasks in parallel computing. The number K of communities may be given by experimenters (Ratio cut, EigenCluster and Normalized Cut) or by referring to the complexity of networks as a whole (Modularity). These criteria forcibly assign every node to one of the communities. However, in social networks, such a forcible assignment is not always preferred since a large fraction of nodes have only few edges and they do not form communities (isolated nodes) [Newman et al. 2002, Zhao et al. 2011] which are hardly dealt with these criteria. Therefore, another group of quality functions are recently gathering interest such as *Graph Scan* [Wang et al. 2008], *Extraction* [Zhao et al. 2011] and ρ -*Dense core* [Koujaku et al.] that are able to deal with such isolated nodes. These criteria find the most dense communities only. Graph Scan and Extraction find only a single community, while ρ -Dense core finds multiple communities at once. In order to extract multiple communities, either of the former two criteria repeats the process in a greedy manner as long as the quality of a newly found community is higher than a pre-determined threshold. The Dense core solves the maximization problem (5) optimally or sub-optimally and excludes automatically isolated nodes. For this reason, we compare these three criteria of Graph Scan, Extraction, and Dense core in the following.

3.2 Framework of change analysis

The framework of change detection on the basis of communities is shown in Fig. 2. We construct the past graph G^- and the present graph G^+ by *double sliding windows*

$$G^- := \hat{G}^{[t-w+1, t]}, \quad G^+ := \hat{G}^{[t+1, t+w]} \quad (7)$$

where w is the window size. From these two graphs, we extract the past communities \mathcal{U}^- and the present communities \mathcal{U}^+ . Then, we calculate a *change score* z^t between \mathcal{U}^- and \mathcal{U}^+ by *Variational Information* (VI) [Meila 2007]:

$$z^t = VI(\mathcal{U}^+, \mathcal{U}^-) = H(\mathcal{U}^+) + H(\mathcal{U}^-) - 2I(\mathcal{U}^+, \mathcal{U}^-), \quad (8)$$

where H and I are the entropy and the mutual entropy of the partition:

$$H(\mathcal{U}) = \sum_{v \in \mathcal{U}} \frac{|\mathcal{U}|}{|\mathcal{V}|} \log \left(\frac{|\mathcal{U}|}{|\mathcal{V}|} \right), \quad (9)$$

$$I(\mathcal{U}, \mathcal{U}') = \sum_{v \in \mathcal{U}} \sum_{v' \in \mathcal{U}'} \frac{|\mathcal{U} \cap \mathcal{U}'|}{|\mathcal{V}|} \log \left(\frac{|\mathcal{V}| |\mathcal{U} \cap \mathcal{U}'|}{|\mathcal{U}| |\mathcal{U}'|} \right). \quad (10)$$

Finally, we regard the time t as a *changing point* if z^t is larger than a threshold θ . There is a trade-off between sensitivity and robustness. For detecting an abrupt change, we need a shorter w , but then we suffer from many false alarms instead. Reversely, with a longer w , we may detect a gradual change, but loose sensitivity. In general, abrupt changes are detected with a small value of w , while gradual changes are found with a large value of w .

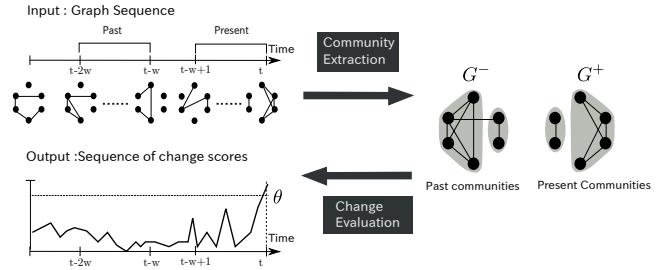


Figure 2: Change detection framework with double sliding windows

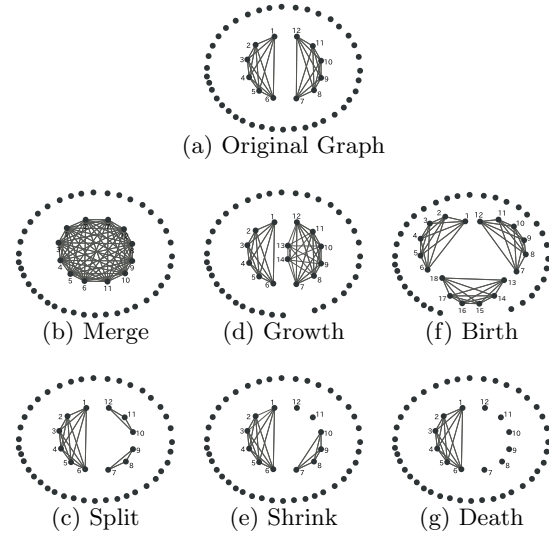


Figure 3: Base graphs of six scenarios of changes

4. EXPERIMENTS

4.1 Analysis of Synthetic graphs

We generated a sequence of 100 graphs $G^{(1)}, G^{(2)}, \dots, G^{(100)}$ based on seven graphs shown in Fig. 3. The original graph Fig. 3 (a) consists of two cliques expressing communities that are surrounded by 50 isolated nodes. From $t = 1$ to $t = 50$, the graph $G^{(t)}$ stays in the original with a flip noise with the probability ϵ . At $t = 51$, the graph generation mechanism changes and starts producing a difference sequence according to one of six scenarios of changes: (b) merge, (c) split, (d) growth, (e) shrink, (f) birth or (g) death.

We say that a detector succeeded if it detected at least one change within $[51 - w + 1, 51 + w]$, otherwise failed. A change was reported when the change score becomes larger than a pre-defined threshold θ . We generated 100 graph sequences in each scenario and evaluated the *Area Under the receive operator Curve* (AUC) for different values of θ . The curve is drawn on the plane of the false alarm rate and the recall rate. After drawing the curve, we calculated the AUC by the trapezoid integration. We set the parameter of community detectors so as to achieve the highest AUCs i.e., we set significance level of Graph Scan and Extraction to 0.01 and the density parameter ρ of ρ -dense core to $\rho = 0.5$. In order to test the performance under difficult conditions, we set the very short window size $w = 5$ and strong noise intensity $\epsilon = 0.1, 0.2$.

Table 2: AUCs of community change detection tasks

	$\epsilon = 0.1$						$\epsilon = 0.2$					
	Merge	Split	Growth	Shrink	Birth	Death	Merge	Split	Growth	Shrink	Birth	Death
Modularity	0.46	0.66	0.77	0.75	0.73	0.76	0.31	0.66	0.68	0.74	0.80	0.72
Graph scan	0.98	0.39	0.96	0.94	0.98	1.00	0.70	0.65	0.62	0.69	0.72	0.65
Extraction	0.57	0.78	0.83	0.78	0.86	0.77	0.54	0.62	0.83	0.67	0.84	0.84
ρ -Dense core ($\rho = 0.5$)	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.82	0.90	0.89	0.97	0.98

Table 3: List of major events in Enron Inc.

ID	Month/Year	Event description
#1	May 00	Significant rise in energy price in California
#2	Aug 00	Enron's stock price attained its greatest value
#3	Jan 01	Electricity crisis led by Enron
#4	Feb 01	Alternation in CEO
#5	Apr 01	CEO gave offensive comments at public
#6	Jul 01	Alternation in CEO
#7	Sep 01	Large amounts of Enron stock was sold
#8	Dec 01	Enron bankrupted
#9	Jan 02	CEO was resigned

Table 2 shows the AUCs of four criteria. Graph Scan gained the high values of AUCs in five scenario with $\epsilon = 0.1$ only, but no with $\epsilon = 0.2$. The ρ -Dense core achieved the highest AUCs in all scenarios of changes both for $\epsilon = 0.1$ and $\epsilon = 0.2$. Extraction showed stable and acceptable performance for both $\epsilon = 0.1$ and $\epsilon = 0.2$.

From these results, we can say that Modularity and Graph Scan are not appropriate for our goal. Therefore, we compared only the ρ -Dense core and Extraction in the following experiments.

4.2 Analysis of E-mail interactions

Next, we examined the network constructed from the daily record [Shetty and Adibi 2004] of e-mail interactions among 151 employees in Enron Inc. A node represents an employee and two nodes are connected at day t when one employee sends an e-mail to another employee in the day. We analyzed the network with ρ -Dense core and Extraction. We trimmed the parameters of community detectors and window size so as to showed the clear peaks in their change scores. The parameter of Extraction and ρ -dense core are set to 0.01 and 0.3, respectively and the window size w is set to $w = 15$. The major social events are summarized in Table 3. In any event, it seems necessary for some persons to communicate to each other.

Fig. 4 shows the change scores of ρ -Dense core and Extraction. The detected change points are shown by red bars. Note that the scales of Y-axis are different. We notice that the change scores of Extraction are larger than those of ρ -Dense core. This indicates that the ρ -Dense core is more pessimistic than Extraction. We examined detail changes at (A) detected by ρ -Dense core. There is no clear change in the amount of e-mails (Fig. 4 (a)) and Extraction does not regard this day as a change point (Fig. 4 (c)). We visualized the graphs before and after this change point (A) in Fig. 5. We can recognize some amount of change in the communities surrounded by the red dotted rectangles.

4.3 Analysis of Phone-call interactions

The network we examined last is constructed from the daily record of phone-call interactions among 106 students

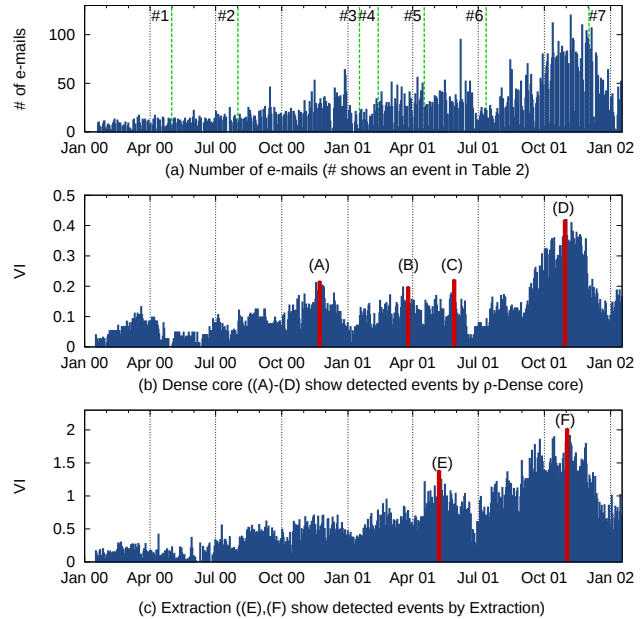


Figure 4: Frequency of e-mails (a), change scores (VI) of ρ -Dense core (b) and Extraction (c) in Enron data. Detected change points are red bars.

Table 4: List of major events in School.

ID	Day/Mon	Related Event
# 1	20 Sep	New semester
# 2	6 Dec	Winter Vacation begins
# 4	17 Jan	Thesis due
# 3	3 Feb	First day of classes

in MIT [Eagle and (Sandy) Pentland 2005]. A node is a student and two nodes are connected if the corresponding two students communicate at day t . We set the window size w to $w = 15$ and the parameter of Extraction and ρ -dense core to 0.01 and 0.2, respectively which were determined as the same with the analysis of E-mail interactions. The major events are summarized in Table 4.

Fig. 6 shows the number of phone-calls and the change scores of two algorithms. The change points detected by the two algorithms are different although (C) and (F) are close. The graphs before/after the change point (A) and (D) are shown in Figs. 7 and 8. We can see that ρ -Dense core finds the changes of many small communities, while Extraction finds the changes of larger communities.

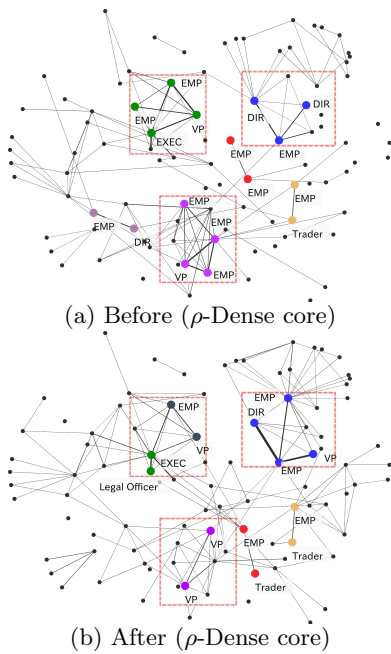


Figure 5: Graphs before/after change point (A) where the width of an edge represents the value of its weight. EMP, MGR, VP, P and DIR stand for Employee, Managers Vice President, President and Director. ρ -Dense core extracts the change of communities surrounded by dotted rectangles. For example, the community in the bottom rectangle seems to have been collapsed after this change.

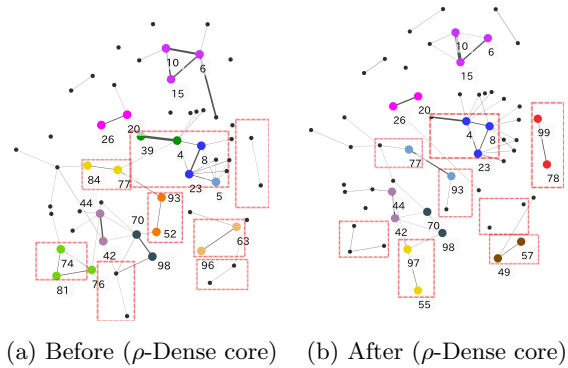


Figure 7: Graphs before/after change point (A) detected by ρ -Dense core. We can see that many small communities have changed.

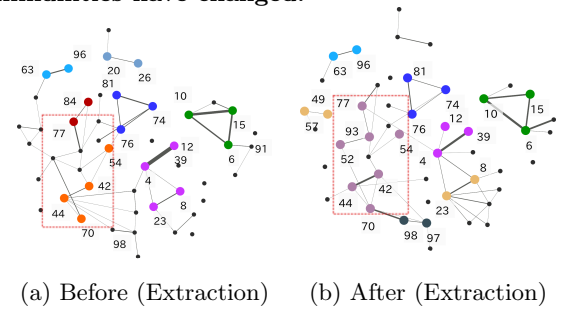


Figure 8: Graphs before/after change point (D) detected by Extraction. Two small communities have been combined into a large community

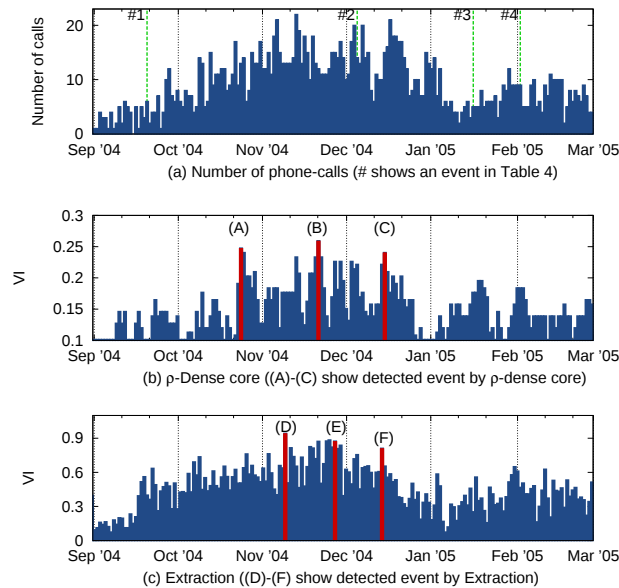


Figure 6: Frequency of phone-calls (a), change scores (VI) of ρ -Dense core (b) and Extraction (c) in MIT student data. The detected change points are red bars.

5. DISCUSSION

The ρ -Dense core model focuses on only the changes in densely connected relatively small subgraphs, rather the changes of the whole graph. This is preferred if such a subgraph corresponds to a small but powerful/special community in practice. In the example of e-mail networks, monitored groups might be a community of executives or a project-based group. In MIT student data, such a community might be a small group of friends or a circle.

We notice that the change points detected by the two algorithms are totally different from the points of the social events in both networks. In some sense, this is natural because these algorithms monitor only relatively small but stationary communities and their changes are not always connected to social events happening in the whole networks. Reversely speaking, that is attraction of these algorithms. That is, from the detected community-level changes, we might find several interesting communities whose change may give a large impact later or show some potential cause of accidents.

ρ -Dense core model has a parameter ρ to determine the standard level of the density of communities to be extracted. A larger value of ρ is useful to find smaller but densely connected communities, and a smaller value is useful for larger but sparser communities. However it is not so easy to find an appropriate value of ρ to the problem at hand. Therefore, a few number of trial and error is necessary while monitoring the change scores and graphs.

6. CONCLUSION

Aiming at community-level change detection in a dynamic network, we have compared the state-of-the-art community extraction algorithms in two viewpoints of the exclusive extraction of stationary communities and the robustness to noise. We demonstrated that our previously proposed community model, ρ -dense core [Koujaku et al.] achieves the best in these two viewpoints. We also have shown that we can detect the several types of changes of stationary communities such as birth, grow, grow, shrink, birth and death. The effectiveness was confirmed on two real-life networks of communications through e-mails and phone-calls.

Future work is to explore more appropriate statistics for community changes and to establish a way to detect the changes skillfully and efficiently, hopefully better than human inspections. More intensive investigation is necessary to confirm the effectiveness in a wider field of real-world applications such as fraud detections and intrusion detections.

7. REFERENCES

- [Agarwal et al. 2008] Nitin Agarwal, Huan Liu, Lei Tang, and Philip S. Yu. 2008. Identifying the influential bloggers in a community. *Proceedings of the international conference on Web search and web data mining - WSDM '08* (2008), 207.
- [Akoglu and Faloutsos 2008] Leman Akoglu and Christos Faloutsos. 2008. Event detection in time series of mobile communication graphs. 1 (2008).
- [Eagle and (Sandy) Pentland 2005] Nathan Eagle and Alex (Sandy) Pentland. 2005. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing* 10, 4 (Nov. 2005), 255–268.
- [Gruhl et al. 2004] Daniel Gruhl, R. Guha, David Liben-nowell, and Andrew Tomkins. 2004. Information diffusion through blogspace. *Proceedings of the 13th conference on World Wide Web - WWW '04* (2004), 491.
- [Hagen and Kahng 1992] Lars Hagen and A.B. Kahng. 1992. New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 11, 9 (1992), 1074–1085.
- [Idé et al. 2007] Tsuyoshi Idé, Spiros Papadimitriou, Michail Vlachos, and T Ide. 2007. Computing Correlation Anomaly Scores Using Stochastic Nearest Neighbors. *Seventh IEEE International Conference on Data Mining (ICDM 2007)* (Oct. 2007), 523–528.
- [Koujaku et al.] Sadamori Koujaku, Ichigaku Takigawa, Mineichi Kudo, and Hideyuki Imai. Dense core model for cohesive subgraph discovery. *Social Networks* (????), (Under Review).
- [Leskovec et al. 2005] Jurij Leskovec, Jon Kleinberg, Christos Faloutsos, H Database Management, and Database Applications. 2005. Graphs over Time : Densification Laws , Shrinking. (2005).
- [Malik 2000] J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 8 (2000), 888–905.
- [Meila 2007] Marina Meila. 2007. Comparing clusterings— an information based distance. *Journal of Multivariate Analysis* 98, 5 (May 2007), 873–895.
- [Newman et al. 2002] M. Newman, Stephanie Forrest, and Justin Balthrop. 2002. Email networks and the spread of computer viruses. *Physical Review E* 66, 3 (Sept. 2002), 035101.
- [Newman 2006] M. E. J. Newman. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E* 74, 3 (Sept. 2006), 036104.
- [Palla et al. 2007] Gergely Palla, Albert-László Barabási, Tamás Vicsek, and A L Barabasi. 2007. Quantifying social group evolution. *Nature* 446, 7136 (April 2007), 664–667.
- [Priebe and Conroy 2006] Carey E Priebe and John M Conroy. 2006. Scan Statistics on Enron Graphs. (2006), 229–247.
- [Sarkar and Boyer 1996] S. Sarkar and K.L. Boyer. 1996. Quantitative measures of change based on feature organization: eigenvalues and eigenvectors. *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1996), 478–483.
- [Shetty and Adibi 2004] J Shetty and J Adibi. 2004. Enron email dataset. *USC Information Sciences Institute, Tech. Rep* (2004).
- [Sun et al. 2007] Jimeng Sun, Christos Faloutsos, Spiros Papadimitriou, and Philip S Yu. 2007. GraphScope. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07*. ACM Press, New York, New York, USA, 687.
- [von Luxburg 2007] Ulrike von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17, 4 (Aug. 2007), 395–416.
- [Wang et al. 2008] Bei Wang, JM Phillips, and Robert Schreiber. 2008. Spatial Scan Statistics for Graph Clustering. *SDM* (2008), 727–738.
- [Zhao et al. 2011] Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. 2011. Community extraction for social networks.. In *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 108. 7321–6.