Community Detection of Time-Varying Mobile Social Networks

Shu-Yan Chan[1], Pan Hui[2], and Kuang Xu[1,3,⋆]

[1] Computer Laboratory, University of Cambridge,
15 JJ Thomson Avenue, CB3 0FD Cambridge, UK
[2] Deutsche Telekom Laboratories / TU Berlin,
Ernst-Reuter-Platz 7, 10587 Berlin, Germany
[3] Department of Electrical and Electronic Engineering,
The University of Hong Kong, Hong Kong
`firstname.lastname@cl.cam.ac.uk`,
`Pan.Hui@telekom.de`

Abstract. In this paper, we present our ongoing work on developing a framework for detecting time-varying communities on human mobile networks. We define the term *community* in environments where the mobility patterns and clustering behaviors of individuals vary in time. This work provides a method to describe, analyze, and compare the clustering behaviors of collections of mobile entities, and how they evolve over time.

1 Introduction

Categorizing mobile objects into communities has numerous potential applications. In Ecology, identifying the animals under observation with their social groups can enhance the study of hunting, mating and other behaviours in context. In online social network studies, grouping individuals into community can help to highlight interaction patterns and identify common attribute amongst individuals. Our target in this paper is human mobile networking.

Previous work has showed that identifying communities can help to improve message forwarding efficiency in Pocket Switched Network (PSN) [1], which is one kind of mobile ad hoc network with intermittent connectivity problem. The improvement is achieved by preferentially forwarding messages to devices that are in the same communities with the messages' destinations.

Our goal is to develop a community detection algorithm that requires little user interventions/adjustments once initialized, and can adapt to the changing and evolving networks. Our work consists of three phases:

1. We first lay a theoretical foundation and formulation for community detection on human mobility traces over a long period of time. This definition is also applicable to community detections in a distributed manner.
2. We evaluate a centralized community detection algorithm which matches our definition of community. (*work in progress*)

⋆ This work begins when the authors were all in Cambridge.

J. Zhou (Ed.): Complex 2009, Part I, LNICST 4, pp. 1154–1159, 2009.

3. We develop and compare the performance of several distributed community detection algorithms. The results from the centralized algorithm serve as an upper bound on how close the communities detected by distributed algorithms can approach the definition.(*future work*)

2 Related Work

Evolution of communities has been well studied in the literature. For example, by utilizing temporal information explicitly, Berger-Wolf *et al.* [2] proposed a framework to identify communities and analyze their evolution in dynamic social networks. Tantipathananandh *et al.* in [3] further propsed a framework for finding communities in social networks that develop over time, and formulated it as a combinatorial optimization problem. They evaluated their algorithms by utilizing several synthetic and real-world datasets of social network. All the above work provides an insightful investigation on the time-varying characteristics in dynamic social networks, but our work differs from them since we focus on the clustering behavior of entities in mobile networks based on human-mobility patterns. By utilizing global time stamp, inter-contact time and contact-duration between mobile devices carried by human in their real lives, we aim at uncovering time-varying communities in dynamic mobile networks which will aid information dissemination [1] in human social life.

Community evolution has also been studied in on-line (Internet) social networks. In [4] [5], communities are identified within some time windows and then merged to reflect its evolution, and heuristics are proposed in those approaches to approximate the optimal solution. Backstrom *et al.* in [6] resorted to several large sources of on-line data which embeds explicit user-defined communities, finding that the inclination of an individual to join a community is affected crucially by the connecting structure of his friends. We do not focus on how the structural features influence the evolving communities in social networks, but study the time-varying communities in human-mobility networks by utilizing temporal information. The work similar to our study is in [7], which took into account of time variability of the information from mobile networks, as in [8], considering a community as a dense connected sub-graph over time, and a node as its member only when it attaches to this sub-graph in a series of time steps. Our study is more ambitious that by incorporating aging factor and history accumulator into the weighted temporal property, we pave the way to detect time-varying communities on mobile network in a distributed way and in real time.

3 Community Detection with Time-Varying Mobility Pattern

A satisfactory community detection algorithm on mobile network should require as few user settings as possible, e.g. a mobile device can be pre-programmed with the algorithm and then distributed to a user and the amount that the user need to fiddle with the algorithm/device's parameters should be minimal, even after the device has been running for a long time and the user may have changed his mobility pattern.

3.1 Definition of Community

Our intuition of a community is a clustering of entities that are "closely" linked to each other, either by direct linkage or by some "easily accessible" entities that can act as intermediates. An entity can belong to more than one community and communities can be hierarchical, such that within a closely connected community there can be sub-communities of which their members have even closer connections between each other.

In our work, we adapted the k-clique [9] community as the basis of our definition of community.

Within a given time period, the *Immediate Neighbours* of an entity n are the set of entities that have "heavy interactions" with n. Some of the Immediate Neighbours of n may be in the same community as n, but it is not necessarily for all of them to be in the same community, for n can be associated with multiple communities. *Immediate Community Neighbours* of an entity n in a community C is thus defined as the set of *Immediate Neighbours* of n that are also in community C.

The concept of "heavy interactions" is subjected to parametrization. In our work, we will explore the contact duration domain and leave other possibility for future works. Within the contact duration domain, the interaction between two entities can be classified as "heavy interaction" if their contact duration exceeds a certain threshold λ. It is the effect of varying the threshold that we wish to investigate.

We redefine k-Clique community in the following way, if entity n is in community C, then there exist at least $k-1$ other entities in it's immediate community neighbours set that all the k entities have "heavy interactions" (above λ) with each other (hence forming a $k-$clique in a graph). Two entities are in the same k-clique community, C, iff their two corresponding nodes are linked by at least one series of adjacent cliques (two k-cliques are adjacent if they share $k-1$ common nodes).

However, over a long period of time, community memberships may be broken or newly formed and communities may evolve. Therefore, if one wants to study the dynamic of communities, it is necessary to partition the trace into smaller time intervals, and applying the community detection algorithm to the data in each smaller time intervals. Ideally, the smaller time intervals should be chosen such that each only cover a period when there is little disturbance to the communities memberships.

However, there is no rigorous definition on what constitute a good partitioning of the time intervals, nor heuristic on how to find "good" partitioning. A "good" partitioning will reveal results that are either desirable (agree with expectation, perhaps inferred by prior knowledge or information) or insightful (reveals communities structures that are not previously known).

Varying λ, α(to be introduced later), and the time intervals might yield different set of communities on a given set of data, thus affecting the *usefulness* of the resulting communities, but there is no mathematical way to differentiate which set of results are better.

3.2 Algorithm

We seek to develop algorithms that can dynamically detect time-varying communities. Our criteria for the algorithm is that it can *adapt* to the change in interaction patterns between the entities and detect the change in communities structures.

Although the final goal in our work is to develop distributed algorithm that will run on pervasive computing devices, in this work-in-progress paper, we present a centralized version of the algorithm. The centralized version provides a upper bound on the communities detected by future distributed algorithms, base on the definition of community in this paper.

The (centralized) algorithm also runs in *real time*, that the communities it detects at any given time are computed solely base on the interaction histories of the entities and have no knowledge of their interactions in the future. This algorithm has its own application, in scenarios where global knowledge of all interactions between entities are accessible as they happen and community information are required on demand. Since we develop this algorithm to facilitate comparison of the distributed version and that it has its own potential applications, we put a bound on the resource it requires - instead of keeping the histories of all interactions between entities (which would have enabled the algorithm to search through all the histories when computing the present communities associations, at a significant increase of computation power and storage requirements), the algorithm keeps a summarized view of the histories (as the distributed version does).

For each pair of entities, there are four variables: a `history accumulator` which is initialized as 0, a current `tally`(of interaction) $tally_t$, an `aging factor` α and a `time interval setting`. The timeline in the environment clock is partitioned according to `time interval setting`. During each time interval, the interactions between the pair(e.g. the contact duration) are tallied. When the environment clock progress to the next time interval, the current tally is "aged" into the `history accumulator` according to the following formula, where the contact duration of the current time interval t is tallied as $tally_t$, w^{ij} represents the `history accumulator` between entities i and j(hereafter in this paper, we will use the graph theory terminologies *nodes* instead of entities, and the w^{ij} is also the *weight* between nodes i and j). w^{ij} takes into account of all the contact histories up to and including the previous time interval using the **Aging Formula**:

$$w_t^{ij} = \frac{w_{t-1}^{ij} * \alpha + tally_{t-1}}{\alpha + 1} \tag{1}$$

and at the first time step, $w_0^{ij} = 0$. It will be shown in later section that the Aging Formula ages the weight in an exponential fashion.

The algorithm allows each pair of nodes to have their own `aging factor` α and `time interval setting`, and they can be updated in response to the change in interaction pattern. However, in this paper, we implement the centralized algorithm using a broadbrush approach that all nodes pairs use the same pre-set α value and use the same `time interval setting` and investigate the effect of changing the α value.

At any environment time, users can request the algorithm to compute for them the current communities structures. They need to specify the k-clique communities they want to detect by specifying the value of k and the contact threshold value, λ. The algorithm will then launch sub-algorithm to detect the k-clique communities. In this paper we implemented the standard CPM $k-clique$ detection algorithm [10].

3.3 Discussion and Preliminary Result

The algorithm needs to be able to *forget* old associations between nodes when they fade away, yet it also need to have some resilience to the temporary fluctuation. The contact history ages with an exponential decay: consider after time t_{end} that there is no further contact between node a and b, then it is straight forward to show that their `history accumulators` age exponentially according to:

$$w_t^{ab} = w_{t_{end}}^{ab} * e^{-\gamma(t-t_{end})} \tag{2}$$

When the aging factor α is less than 1, the history ages rapidly (since for each new time step, the influence of the entire contact history before is less than that of the new contact statistic); however, if the aging factor is allowed to be set to be greater than 1, the resulting edge weights would be more resilience to temporary fluctuation as more emphasis is placed on the history than the current contact statistic, yet, over time, the influence of past contact counts would still fade away. This can therefore offset some of the impacts from setting a sub-optimal time step length while still allowing aging to take place.

Fig. 1 shows result of running our algorithm against an artificial mobility traces which is generated in such a way that there is a sudden change in mobility pattern between time units 4 and 5 on the x-axis. The y-axis denotes the classic Jaccard index [11] when the communities detected in one snapshot is compared to those detected in the previous snapshot. It is clearly shown that the algorithm managed to adapt to change in clustering behaviour occur at around time index 4 and 5 (the dip of lower similarity values around that time indicate a bigger change in detected communities) , with different settings of aging factor α. The figure also shows that the algorithm adapts slower with a smaller α value.

However, natural and social systems are inherently noisy, therefore, in future work we will experiment on relaxing the requirement that all pairs in a clique need to be linked. We would consider a group of nodes as $clique_{relax}$ if they are just missing a few links to form a proper clique (this is a more relax condition than those found in the study of clique community on weight network ,$CPMw$ [10]). In a clique, the distance between two nodes is 1 (direct connection), however, when one of the link is removed, the maximum distance between two nodes

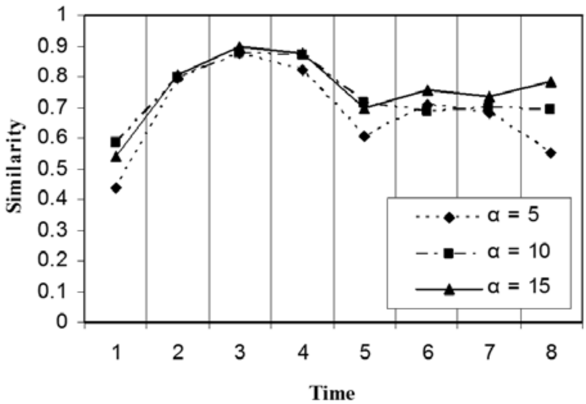

Fig. 1. The effect of community detection with different Aging factor (α)

in that clique becomes 2. If we restrict ourselves so that the maximum number of links that can be dropped to be less than $k-1$, then we ensure that all nodes pairs are still at least indirectly connected and the maximum distance are still 2. We call such definition of k-clique as k-clique$_{relax}$. A community is equivalent to a maximal set of k-clique$_{relax}$, with edge weight higher than λ, that can be reached from each other via series of k-clique$_{relax}$ adjacency connections.

References

1. Hui, P., Crowcroft, J.: Bubble Rap: Forwarding in small world DTNs in ever decreasing circles. Technical Report UCAM-CL-TR-684, University of Cambridge, Computer Laboratory (2007)
2. Berger-Wolf, T.Y., Saia, J.: A framework for analysis of dynamic social networks In: KDD 2006 Philadelphia, USA (2006)
3. Tantipathananandh, C., et al.: A framework for community identification in dynamic social networks. In: KDD 2007, San Jose, California, USA (2007)
4. Chakrabarti, D., Kumar, R., Tomkins, A.: Evolutionary clustering. In: KDD 2006, Philadelphia, USA (2006)
5. Tantipathananandh, C., et al.: Structural and temporal analysis of the blogosphere through community factorization. In: KDD 2007, San Jose, California, USA (2007)
6. Backstrom, L., Huttenlocher, D., Kleinberg, J.: Group formation in large social networks: membership, growth, and evolution. In: KDD 2006, Philadelphia, USA (2006)
7. Scherrer, A., et al.: Description and simulation of dynamic mobility networks. Comput. Netw. 52(15) (2008)
8. Scherrer, A., et al.: Synchronization reveals topological scales in complex networks. Physical Review Letters 96 (2006)
9. Palla, G., et al.: Uncovering the overlapping community structure of complex networks in nature and society. Nature 435(7043), 814–818 (2005)
10. Farkas, I., Abel, D., Palla, G., Vicsek, T.: Weighted network modules. New Journal of Physics 9 180 (2007)
11. Jaccard, P.: Bulletin de la Societe Vaudoise des Sciences Naturelles 37, 547 (1901)