International Journal of Current Engineering and Technology
ISSN 2277 - 4106
© 2013 INPRESSCO. All Rights Reserved.
Available at http://inpressco.com/category/ijcet

Research Article

Estimation of new Similarity Measures for existing frameworks over Time for tracking Community structure in Online Social Network

Sanjiv Sharma[a*] and G.N. Purohit[a]

[a]Department of computer Science Banasthali Vidyapith , Bansthali Rajasthan(INDIA)

Accepted 2 Jan. 2013, Available online 1March 2013, **Vol.3, No.1 (March 2013)**

Abstract

Many real-world social networks are intimately organized according to a community structure. Most social networks are dynamic and connections between people change naturally overtime. Researchers have begun to consider the problem of tracking the formation of groups of users in social network. The situation is complicated by the fact that subgrou ps may split or merge, so that cohesiveness is not necessarily a property of a single subgroup, but may sometimes relate to a family of one or more related subgroups. However, in general, cohesive families of subgroups at one time period should be similar to corresponding subgroups at a different time period. Similarity is a topic that has received attention in a wide variety of scientific fields and a number of approaches are available for the measurement of similarity. This paper describes an efficient way for finding similarity in subgroups or clusters and tracking community which persist over time in dynamic networks.

Keywords*: social network, community, similarity, groups.*

1. Introduction

A social network is simply a structure consisting of people or other entities embedded in a social context, with a relationship among those people that represent interaction, collaboration, or influence between entities. The extreme popularity and rapid growth of these online social networks represents a unique opportunity to study, understand, and leverage their properties. A community (Tantipathananandh et al, 2007) is intuitively understood as a set of entities where each entity is closer, in the network sense, to the other entities within the community than to the entities outside it. Therefore, communities are groups of entities that presumably share some common properties and play similar roles within the interacting phenomenon that is being represented.

This paper describe a similarity measure (Gregson et al, 1975) for tracking the evolution and structure of communities in multiple snapshots of a dynamic network (Greene et al,2010), where the life-cycle of each community is characterised by a series of significant events. Based on this model, we propose a simple but effective method for efficiently identifying and tracking these dynamic communities (F. Radicchi et al,2004), which involves matching communities found at consecutive time steps in the individual snapshot graphs. Unlike other approaches, the method is independent of the choice of underlying community finding algorithm which is applied to the individual step graphs. It can also aggregate information from either disjoint or overlapping groupings (McDaid et al,2010) of nodes. To evaluate the method, we introduce a procedure for generating synthetic dynamic networks.

2. Related Work

Cohesive subgroups should have a core group of people that remain the same over different time periods. The situation is complicated by the fact that subgroups may split or merge, so that cohesiveness is not necessarily a property of a single subgroup, but may sometimes relate to a family of one or more related subgroups. However, in general, cohesive families of subgroups at one time period should be similar to corresponding subgroups at a different time period. Similarity is a topic that has received attention in a wide variety of scientific fields and a number of approaches are available for the measurement of similarity.

Mathematically, similarity may be viewed as a geometric property involving the scaling or transformation necessary to make objects equivalent to each other. Similarity can be defined as the inverse of distance, with a well-known distance measure being Euclidean distance (Chi et al,2007), which itself is a special case of a family of distance measures known as Minkowski metrics. However, distance measures typically require a vector (spatial) model of the entities being compared, which is often not appropriate for comparing aggregations of nodes in a network. In developing methods to assess the

* Corresponding author: Sanjiv Sharma

similarity between different species, numerical taxonomists have developed and utilized a number of similarity measures. Many of these measures involve some sort of correlation, a construct that is conceptually related to similarity. One correlation measure is the cosine distance or dot product that measures the angle between two objects represented as vectors of numerical features. However, since features cannot always be expressed on a well-defined numerical scale, researchers (e.g., psychologists) have developed feature models of similarity that assess similarity based on a comparison of matching and mismatching features, using a set-theoretic approach .

Tversky's feature contrast model (Tversky et al,1977) expressed the degree of similarity of two stimuli to a linear combination of their common and distinctive features. Gregson recommended a content similarity model where similarity was expressed as the ratio of the intersection of the features for the objects being compared to the union of their features. A simplified version of the content similarity model is the Jaccard coefficient(Jaccard et al,1901) (first proposed in 1901 by Paul Jaccard), which is defined as the size of the intersection divided by the size of the union of the objects being compared.

Johnson [8] proposed the ultra metric distance as a way of measuring distance within a hierarchy. For comparing two different clustering hierarchies, one heuristic method for estimating similarity consists of converting each hierarchy to a matrix of ones and zeros where the ones represent the parent-child links in each hierarchy.

The similarity between two hierarchies is then estimated as the correlation between the two corresponding matrices of ones and zeroes. A more formal approach is to use quadratic assignment to assess the similarity between two partitions. Quadratic assignment is a combinatorial approach, where simulation is used to create a sampling distribution of possible shuffles of a partition in terms of a correlation or regression statistic between the original partition and each shuffled version. The similarity observed between two partitions is then compared with that sampling distribution to see how extreme or notable the observed statistic actually is. Other related work by Falkowski focused on finding community instances using similarity.

In SCAN Social Cohesion Analysis of Networks (Chin et al,2009)previous two steps (Select and Collect) can be repeated to discover candidate cohesive subgroups at any desired point in time. If subgroupings are cohesive they should tend to change less over time. Thus, subgroupings that maintain cohesion over time should be preferred over those that do not.

The second insight is that high levels of cohesion are unlikely to arise by chance. According to Social Identity Theory, group members feel closer if they are similar to each other (A. Clauset et al,2004). Thus, capitalizing on the best betweenness cutoff to create the most cohesive subgroupings, is unlikely to distort the patterns (Wang H et al,2002) that exist in the data. Sub graphs is chosen and calculated. The chosen cohesive subgrouping is then based on the cutoff betweenness centrality that maximizes the similarity of subgroupings obtained in adjacent time periods.

Using a set-theoretic approach (Memon et al,2008), similarity can then be characterized as the ratio of the set intersection of subgroups to the corresponding set union. In practice, this is equivalent to the proportion of the number of common pairs in all clusters in both time periods relative to the total number of possible combinations of pairings. In developing a similarity measure, three issues need to be considered. First, the selection of the time periods or windows to be used for comparison, second, the comparison of time windows, and third, the network dynamics.

For the selection of time periods, the approach used in this paper is to select a similarity measure based on static time windows for the entire time analysed to simplify the research problem, in order to see how it works. The approach used in this paper could then be extended to varying time windows and the similarity measure would be modified accordingly after relaxing the view of persistence over time. Further analysis could be carried out for comparing participants for candidate subgroups at a particular time period with other time periods using a multi-window approach.This would involve changing the Select criteria to handle multiple windows. In terms of network dynamics, the present approach considers only networks where membership is either fixed or else where new members enter, but existing members do not leave the network (and subgroups) at different time periods. Thus, the implementation of the SCAN method considered here does not consider members leaving the network orboth entering and leaving. These cases are left for future work.

Based on the assumed network dynamics noted above, two similarity-based approaches were developed for finding cohesive subgroups. In the first approach, cohesion was examined across all subgroups where there was constant membership in subgroups over the adjacent time periods. In the second approach, cohesion was examined for the largest subgroup (only) in the first of the two time periods being compared. Both similarity measures are potentially useful, but different ways to measure cohesion, where the first takes into account constant membership of members in subgroups in adjacent periods of time, and the second takes into account incoming actors into the subgroups.

For the first similarity approach, the cohesion across all subgroups between two consecutive time periods T1 and T2 can be calculated according to equation 1:

$$SimT1,T2 = 2 * N(T1 \cap T2)/N(T1 \cup T2) \quad \ldots\ldots\ldots \quad (1)$$

where N(T1 ∩ T2) is the number of pairs where both members are in the same clusterin T1 and T2 and N(T1 UT2) is the total number of pairs who are in the same cluster in either (or both) T1 and T2. A factor of 2 is added as a multiplier to the numerator of this expression so that the resulting similarity measure is normalized between 0 and1.

The second similarity approach measures the cohesion of the largest individual subgroup. This examines all the possible pairwise relationships between members of the subgroup, and determines how many of the pairs still exist (i.e., are inside a subgroup)in the second time period. The

similarity can then be calculated using the following formula according to equation 2.

$SimT1,T2=N(S1 \cap T2)/N(S1)$ …….. (2)

where S1 is the largest subgroup in T1, N(S1 ∩ T2) is the number of common pairs fromthe largest subgroup S1 that still exist in T2, and N(S1) is the number of pairs in thelargest subgroup S1.

Limitation: a third similarity measure would need to be defined that considers both new members and members that leave the subgroups, since the two similarity measures currently defined only take into account constant membership and new membership. This view that members will be persistent in the same subgroups over different periods of time can then be relaxed and the effect of varying duration of time windows for each time period can be studied where the network is changing in size and composition as members join and leave the network. Subsequent similarity equations should also take into account the variability of the time window, which provides considerable scope for trying out more comprehensive schemes for assessing similarity over time that might incorporate a variety of techniques, including time-series data analysis, temporal algorithms, and sliding window algorithms.

3. Proposed Work

The paper represent a Social network as a set of t time graphs {g1, . . . , gt}, providing snapshots of the nodes and edges in the overall network at successive intervals. The problem then becomes the identification of a set of k communities D = {D1, . . . ,Dk} that are presenting the network across multiple time steps. We refer to step communities that are identified at individual time steps, which represent specific observations of dynamic communities at a given point in time. Unlike the approach described by [20], these need not necessarily comprise of cliques. Rather, the observations can be taken from any disjoint or overlapping grouping that provides assignments for some or all of the nodes in the complete network. We denote the set of k_t step communities or cluster identified at time t as Ct = {Ct1, . . . ,Ctkt}.

In the context of the model described above, a key question concerns how best to map step communities at each time t to the existing set of dynamic communities D. Further questions may arise regarding the feasibility of performing this correspondence process in an efficient manner for graphs containing a large number of nodes and communities. The first step grouping C1 is generated by applying a chosen static community finding algorithm to the graph G1 – we use this graph to bootstrap the process. A distinct dynamic community is created for each static community. The next grouping C2 is generated on the graph g2.

To perform the actual matching between Ct and the fronts {F1, . . . , Fk0}, we employ the widely adopted Jaccard coefficient for binary sets (Jaccard, 1912). The Jaccard distance, which measures dissimilarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1, or, equivalently, by dividing the difference of the sizes of the union and the intersection of two sets by the size of the union: Given a static community C_{ta} and a recent community Ri, the similarity between the pair is calculated as:

$sim(Cta, Ri) = Cta \cap Ri / Cta \cup Ri$ …….. (3)

The second similarity approach measures the cohesion of the largest individual subgroup. This examines all the possible pairwise relationships between members of the subgroup, and determines how many of the pairs still exist (i.e., are inside a subgroup) in the second time period. The similarity can then be calculated using the following formula according to equation 4:

$sim(Si,Ri)=Cta \cap Ri/Si$ …….. (4)

Si is number of pairs in the largest subgroup.

Propose similarity measure need to be defined that considers both new members and members that leave the subgroups, since the two similarity measures currently defined only take into account constant membership and new membership.The Jaccard distance, which measures dissimilarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1, or, equivalently, by dividing the difference of the sizes of the union and the intersection of two sets by the size of the union:

$sim(Cta, Ri) = ((Cta \cup Ri)-(Cta \cap Ri)) / Cta \cup Ri$ …… (5)

If the similarity exceeds a matching threshold $T \in [0, 1]$, the pair are matched and Cta is added to the timeline for the dynamic community Di. For practical purposes, the intersection calculations required for Eqn. 1 can be performed efficiently using a number of strategies, including optimizations based on sorted sets [18], or bit array operations [19]. In the implementation used in this paper, we represent dynamic communities in terms of a node-community map against which incoming step communities are compared. This change leads to substantial performance improvements when compared to a naive implementation based on pairs of set structures. If no suitable match is found for Cta above the threshold T, a new dynamic community is created for Cta. An outline of the entire process is provided as per follows:

1. Calculate Eigen value centrality and generate important nodes.
2. Extract cluster or static community $C_1$of social network G_1using spectral clustering.
3. Initialize D by creating a newdynamic community for each static community $c_{1i} \in C_1$.
4. For each subsequent step t > 1, extract C_t from G_t.
5. Process every $C_{ta} \in$ Ct as follows:
    - Match all dynamic communities Di for which $sim(C_{ta}, Ri) > T$ or $sim(Si,Ri) > T$.

- If there are no matches, create new dynamic community containing C_{ta}.
- Otherwise, add C_{ta} to each matching dynamic community.

6. Update the set of current community Ri for each dynamic community to be the latest matched static community. For each case where one existing dynamic community has been matched to 2 or more static communities, create a split dynamic community.
7. Repeat from #2 until all time graphs have been processed.

4. Experiment

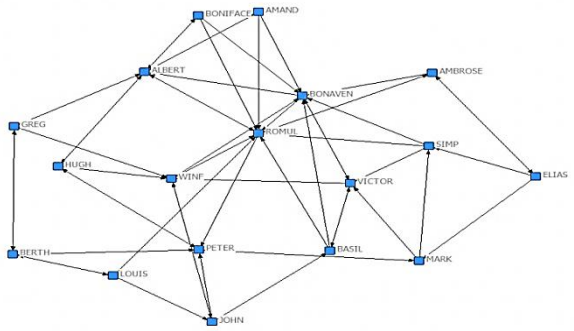
The goal here was to determine whether applying a step-based dynamic community finding process could improve ability to detect dynamic communities, when compared with traditional static community finding methods which treat dynamic networks as a single graph without regard to temporal information. Consider a social network data set, Sampson recorded the social interactions among a group of monks while resident as an experimenter on vision, and collected numerous socio metric rankings. During his stay, a political "crisis in the cloister" resulted in the expulsion of four monks (Nos. 2, 3, 17, and 18) and the voluntary departure of several others - most immediately, Nos. 1, 7, 14, 15, and 16. (In the end, only 5, 6, 9, and 11 remained). Most of the present data are retrospective, collected after the breakup occurred. They concern a period during which a new cohort entered the monastery near the end of the study but before the major conflict began. The exceptions are "liking" data gathered at three times: SAMPLK1 to SAMPLK3 - that reflect changes in group sentiment over time (SAMPLK3 was collected in the same wave as the data described below). Information about the senior monks was not included.

Four relations are coded, with separate matrices for positive and negative ties on the relation. Each member ranked only his top three choices on that tie. The relations are esteem (SAMPES) and disesteem (SAMPDES), liking (SAMPLK) and disliking (SAMPDLK), positive influence (SAMPIN) and negative influence (SAMPNIN), praise (SAMPPR) and blame (SAMPNPR). In all rankings 3 indicates the highest or first choice and 1 the last choice. (Some subjects offered tied ranks for their top four choices).

Data:

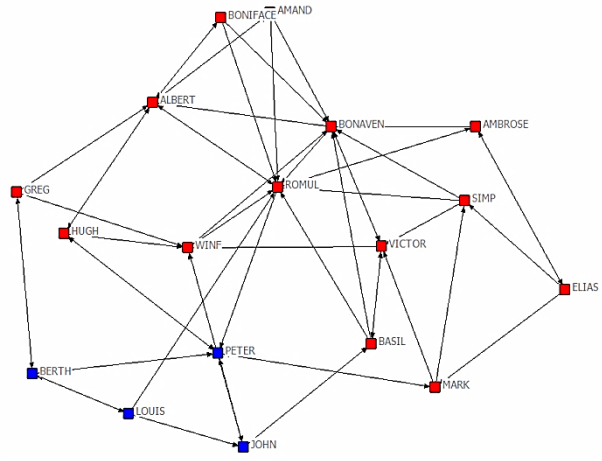
ROMUL AMBROSE 2
ROMUL PETER 3
ROMUL ALBERT 1
BONAVEN ROMUL 3
BONAVEN VICTOR 2
BONAVEN ALBERT 1
AMBROSE ROMUL 2
AMBROSE BONAVEN 3
AMBROSE ELIAS 1
BERTH PETER 3
BERTH LOUIS 1
BERTH GREG 2
PETER BERTH 3
PETER HUGH 2
PETER MARK 1
LOUIS ROMUL 1
LOUIS BERTH 3
LOUIS JOHN 2
VICTOR BONAVEN 2
VICTOR WINF 1
VICTOR BASIL 3
WINF ROMUL 3
WINF BONAVEN 2
WINF JOHN 1
JOHN PETER 2
JOHN WINF 3
JOHN BASIL 1
GREG BERTH 3
GREG WINF 1
GREG ALBERT 2
HUGH PETER 3
HUGH WINF 1
HUGH ALBERT 2
BONIFACE ROMUL 3
BONIFACE BONAVEN 2
BONIFACE ALBERT 1
MARK PETER 2
MARK VICTOR 1
MARK SIMP 3
ALBERT ROMUL 3
ALBERT HUGH 1
ALBERT BONIFACE 2
ALBERT AMAND 2
AMAND ROMUL 3
AMAND BONAVEN 2
AMAND ALBERT 1
BASIL ROMUL 1
BASIL BONAVEN 2
BASIL VICTOR 3
ELIAS AMBROSE 3
ELIAS MARK 2
ELIAS SIMP 1
SIMP ROMUL 2
SIMP BONAVEN 3
SIMP VICTOR 1

Above dataset a analysed by UCINET in form of graph G, in which all actor represented by vertex V is communicate to each other by edges E is analysed by UCINET in time t1 , time t2 and find the structure of community in following way:



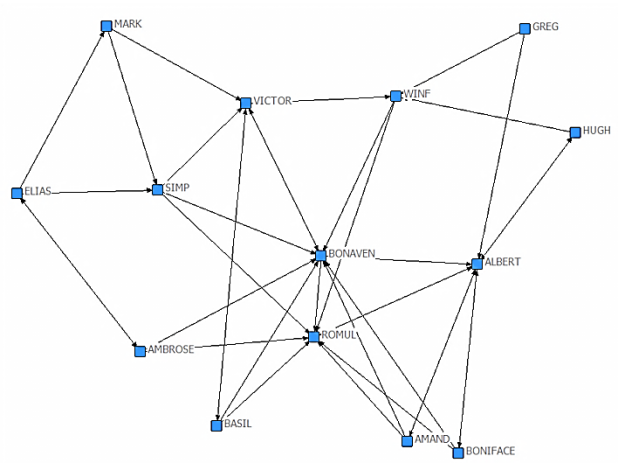
Graph G1

Step 1: Calculate Eigen value centrality and generate important nodes subsequently Extract cluster or static community C of social network g_1 using spectral clustering. In Graph G2 red vertices shows static community and blue vertices is not a part of community.



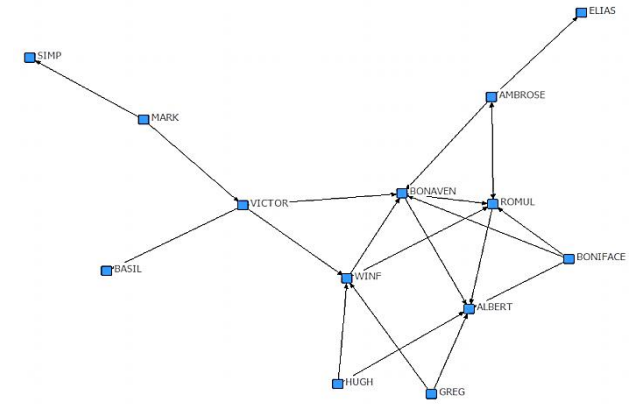
Graph G2

Step 3: In time t1 static community C1 is created and generate Graph G3. 3. Initialize D by creating a new dynamic community for each static community $c1i \in C1$



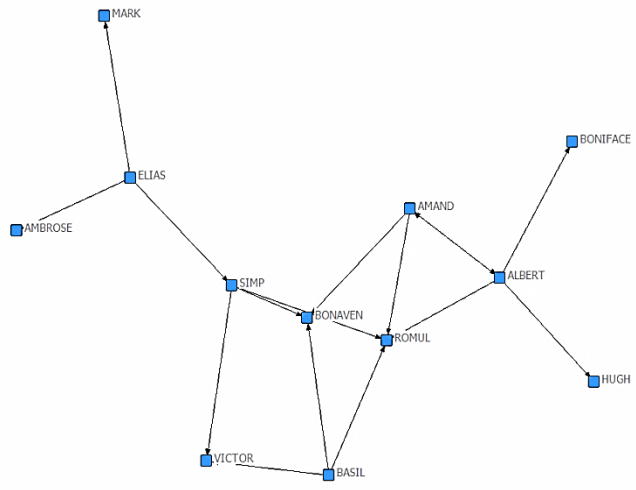
Graph G3

Step 4: In time t2 static community C2 is created and generate Graph G4.



Graph G4

Step 5: Update the set of current community Ri for each dynamic community to be the latest matched static community. For each case where one existing dynamic community has been matched to 2 or more static communities, create a split dynamic community. Resultant tracked dynamic community shows in Graph G5



Graph G5

5. Conclusion

In this paper, we have described effective method for finding similarity and tracking communities in dynamic networks. Analysis of social network dataset shows that the proposed method performs better than traditional static community finding strategies which do not take temporal information into account. Additionally, we have proposed a third similarity measure would need to be defined that considers both new members and members that leave the subgroups, since the two similarity measures currently defined only take into account constant membership and new membership , which was limitation of previous approach.

References

Tantipathananandh, C., Berger-Wolf, T. & Kempe, D. (2007). A framework for community identification in dynamic social networks. In Proc. 13th ACM SIGKDD, *International conference on Knowledge Discovery and Data mining*, 717–726, ACM.

Greene,D.,Doyle,D.& Cunningham,P.(2010) .Tracking the evolution of communities in dynamic,Social networks. In Proc. International Conference on Advances in Social Networks Analysis and Mining(ASONAM'10),*IEEE*.

Chi, Y., Song, X., Zhou, D., Hino, K. & Tseng, B. (2007). Evolutionary spectral clustering by incorporating temporal smoothness. In Proc. 13th ACM SIGKDD, *International conference on Knowledge Discovery and Datamining*, 153–162,ACM.

Gregson AMR (1975) Psychometrics of similarity. *Academic*, NY, USA

Elmore LK, Richman BM(March 2001) Euclidean distance as a similarity metric for principal component analysis. *Month Weather Rev* 129(3):540–549

Tversky A (1977) Features of similarity, *Psychol Rev* 84(4):327–352

Jaccard P (1901) Distribution de la flore alpine dans le bassin des dranses et dans quelques rgions voisines. *Bulletin del la Socit Vaudoise des Sciences Naturellese*, 37:241–272

Johnson CS (1967) Hierarchical clustering schemes, *Psychometrika*, 32

Falkowski T, Bartelheimer J, Spiliopoulou M (2006) Community dynamics mining. In: Proceedings of 14th *European conference on information systems*(ECIS2006).Gteborg,Sweden

Wang H, Wang W, Yang J, Yu SP (2002) Clustering by pattern similarity in large data sets. In: SIGMOD '02: Proceedings of the2002ACM SIGMOD international conference on management of data. *ACM*, New York, NY, USA, pp 394–405

Ebner, W.; Leimeister, J. M.; Krcmar, H. (2009): Community Engineering for Innovations - The Ideas Competition as a method to nurture a Virtual Community for Innovations. In: *R&D Management*, 39 (4),pp 342–356

McDaid, A. & Hurley, N. (2010). Detecting highly overlapping communities with Model-based Overlapping Seed Expansion. In Proc. International Conference on Advances in *Social Networks Analysis and Mining* (ASONAM'10), 112–119, *IEEE*.

Baumes, J.; Goldberg, M.; and Magdon-Ismail, M. 2005.Efficient identification of overlapping communities. *Intelligence and Security Informatics* (ISI) 27–36.

F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi(2004). Defining and identifying communities in networks, *Proc Natl Acad Sci USA*, 101(9):2658–2663.

A. Clauset, M. E. J. Newman, and C. Moore(2004). Finding community structure in very large networks,*Physical Review* E, 70(6):066111.

Chin A (January 2009) Social cohesion analysis of networks: a method for finding cohesive subgroups in social hypertext. PhD thesis, *University of Toronto*.

Memon N, Harkiolakis N, Hicks LD (2008) Detecting high-value individuals in covert networks: 7/7 London bombing case study. In Proceedings of the 2008 IEEE/ACS International Conference *on computer systems and applications. IEEE Computer Society*, *Washington* DC,USA, 4–31 April 2008, pp 206–215

Baeza-Yates, R. (2004). A fast set intersection algorithm for sorted sequences. In Proceedings of the 15th *Annual Symposium on Combinatorial Pattern Matching (CPM* 2004), vol. 3109, 400–408, Springer.

Asur, S., Parthasarathy, S. & Ucar, D. (2007). An event-based framework for characterizing the evolutionary behaviour of interaction graphs. In Proc. 13th *ACM SIGKDD international conference on Knowledge Discovery and Data mining*, 921, ACM.

Palla, G., Barab´asi, A.-L., and Vicsek, T. (2007). Quantifying social group evolution. *Nature*, 446:664–667.