Top-K Strongest Strength Communities in Dynamic Networks

Sushama Patil, Prof.Mansi Bhonsle
Computer Science and Engineering Dept.
GHRCEM, Wagholi, Pune, India

Abstract—Today most of the networks are dynamic in nature such as Biological network, Social networks like Facebook, Twitter etc. These networks contain a characteristic called as communities which are group of vertices having strong internal connections and weak external connections. Finding such communities and analyzing their progression from one timestamp to another is essential trend in data mining now days. This paper presents a two stage framework for community detection and community strength analysis in directed dynamic networks. Finding the evolution of community strength is helpful to understand the underlying behavior of community. This framework is also useful to find out top k strongest communities in directed dynamic networks. One can also find out what will be the change in community strength from one timestamp to next.

Index Terms— Dynamic networks, community, strength, data mining, progression, timestamp, smoothness parameter, strongest, progression net, PACS algorithm, community group, NMF, strength transmission net, strength reception net.

1 Introduction

Dynamic networks are used to represent the time based behavior of many complex systems in real world. Generally graph theory is used to model these dynamic networks [1]. Fundamental parts in the network are represented as nodes and interaction between them as links. One of the most important aspects of dynamic network is the identification of communities which are collections of individuals who interact frequently. However, all these detected communities are stationary and isolated at a specific timestamp. Thus we are unaware about when these communities were formed or when they are going to split. Community strength is a time based quantity which gives the probability that a particular community has a constant membership at current timestamp. Its value may change as network progresses.

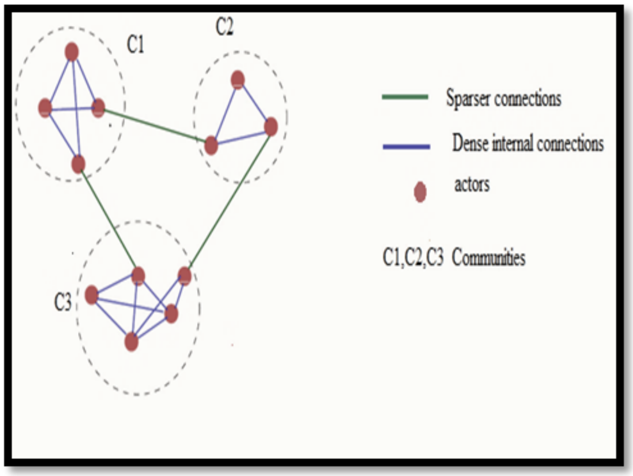

Fig.1. Community Structure

If community strength of any community is high then that community has little possibility of changing its members i.e. adding new members and leaving existing members [2]. It is helpful to find out some interesting community information which cannot be directly obtained from traditional community analysis. By using the value of community strength one can find out stable communities and track it over an entire observation period. In Fig 1 there are three communities namely c1, c2, c3. There are dense internal connections between actors of same community. There are sparser connections between actors of two different communities. The main goal of our proposed method is finding out what will be the effect of link direction on change of community strength from one timestamp to next because directedness is an important feature of many real networks. Specifically ignoring link direction when observing for communities may result into incomplete or misleading results.

2 Related Work

Many methods have proposed for community detection till date. M. E. J. Newman and M. Girvan [7] presented divisive algorithms for discovering community structures in network also proposes a new method to determine the strength of the community structure, which gives an objective metric for choosing the number of communities into which a given network should be divided. It is having high computational complexity. For systems having large number of vertices it becomes intractable.M. Gupta, J. Gao, Y. Sun, and J. Han
[5] introduced the concept of evolutionary outliers with respect to hidden evolving communities, i.e., Evolutionary Community Outliers (ECOutliers). These outliers correspond to the objects which go against the common evolutionary trend that majority of the objects in a community follow, and the trend must be obtained from community matching. Here the objective is to minimize community matching error, in which the contributions from outlier objects should be lower. This proposed algorithm focuses only on two timestamps, so thedrawback of this is that it cannot handle multiple timestamps.

N. Du, J. Gao, and A. Zhang [2] introduced a new method for analyzing the progression of community strengths. Community strength is a sequential measure which represents the probability that a particular community has a firm structure at the current timestamp. A two-stage framework is proposed which includes community detection and community strength analysis. This method can provide reliable and consistent community strength and also less sensitive to short-term noises in the current network. It considers only undirected network not directed network. It does not measure the impact and consequently the change in community strength based on immediate preceding timestamps.

Falkowski, T., Bartelheimer, J., Spiliopoulou, M [4] projected a method for tracking the evolution of communities. This applies to the general case of random graphs. The method contains first performing ordinary community detection on time timestamps of the network by maximizing modularity. A graph of communities detected at each time step is then created, and meta-communities of communities are detected in this graph to match communities over time. The main drawback of this method is that no temporal smoothing is used, so the identified communities are possibly to be unstable.

Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng [6] introduced new method to discover communities from social network data and to analyze the community evolution. An innovative algorithm *FacetNet* for analyzing communities and their evolutions through a robust *unified* process is proposed. This algorithm deviates from the traditional two-step approach to analyze community evolutions. It proposes only soft modularity and also expensive. The given method only considers the link information not the content information. P. Brodka, S. Saganowski, and P. Kazienko [3] discovered the method for detecting the evolution of communities. This method finds out the changes like merging, splitting and surviving between successive communities. However, the information delivered by this is restricted to only adjacent timestamps which are unable to give us a complete representation of the community evolution.

3 Methodology

The system architecture of the proposed framework is shown in Fig 2. This framework consists of two stages: **Community Detection** which is the method for dividing the network from each timestamp into communities and **Community Strength Analysis** which is the method for analyzing the strength of each community over entire period.

A. Community Detection at Each Timestamp using NMF Technique-

Given a series of directed networks $G^t = (V, E^t, W^t)$ $(1 \leq t \leq T)$, each network is first divided into K_t communities at each timestamp t. For this purpose Asymmetric Non-Negative Matrix Factorization (NMF) technique is used [8]. There are various reasons to use it: 1) It is useful for both hard and soft clustering. 2) Due to its non-negative constraints it is having high interpretability that reveals underlying behavior of communities which is useful to study the progression of community strength in directed network.3) It is capable of dealing with overlapping communities. So each timestamp network is factorized as follows:

$$\min_{C^t, S^t > 0} W^t - C^t S^t C^{t^{Trans}} \tag{1}$$

Here W^t is an $N \times N$ asymmetric matrix representing the relationship between entities at timestamp t, C^t is $N \times K$ Community Indicator matrix where K is number of communities at a timestamp t and communities may be overlapping or non-overlapping. Here If node i is allocated to community k at timestamp t, then C^t_{ik}= 1 otherwise 0. All such community indicator matrices for all timestamps are placed in Community Group ^C and S^t is $K \times K$ Community relationship matrix where each S^t_{ij} represents similarity between community i and community j that are detected at timestamp t.

C^t and S^t are iteratively updated as follows:

$$C^t_{ij} \leftarrow C^t_{ij} \sqrt[4]{\frac{(W^{t^{Trans}} C^t S^t + W^t C^t S^{t^{Trans}})_{ij}}{(C^t S^t C^{t^{Trans}} C^t S^{t^{Trans}} + C^t S^{t^{Trans}} C^{t^{Trans}} C^t S^t)_{ij}}} \tag{2}$$

$$S^t_{ij} \leftarrow S^t_{ij} \frac{(C^{t^{Trans}} W^t C^t)_{ij}}{(C^{t^{Trans}} C^t S^t C^{t^{Trans}} C^t)_{ij}} \tag{3}$$

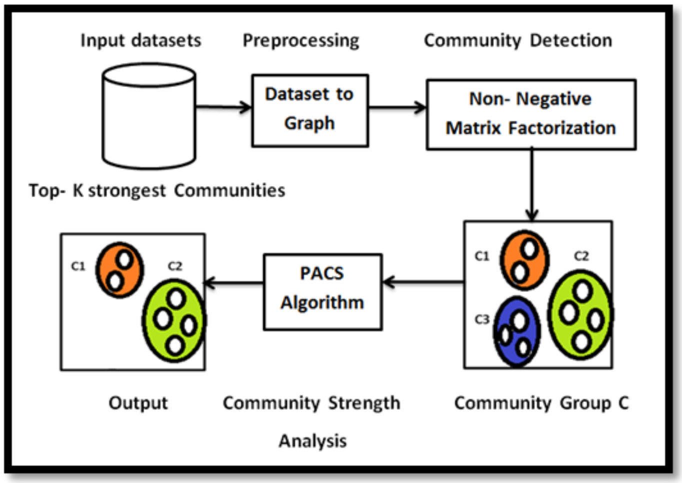

Fig.2 System Architecture

B. Community Strength Analysis using PACS algorithm-

After detecting communities from all timestamps and putting them in community group ^C, strength of each community a_{zt} is calculated based on following Eq. 4

$$a_{zt} = \frac{[\alpha C_z{}^{Trans} (W^t - D^t) C_z + (1-\alpha) C_z{}^{Trans} (W^{t-1} - D^{t-1}) C_z] \mu_t}{\sum_{z=1}^{K} [\alpha C_z{}^{Trans} (W^t - D^t) C_z + (1-\alpha) C_z{}^{Trans} (W^{t-1} - D^{t-1}) C_z]} \tag{4}$$

where W represents sum of weights for network . D equals to dd^{Trans} where d is an $N \times 1$ vector and each d_i is the degree of node i. The time based community strength should depend on the current network, and it would not diverge too intensely from the preceding timestamp's network. So smoothness parameter μ_t which controls the whole community strength at a specific timestamp t is used. This smoothness is applied on time based community strength instead of applying it between the communities detected in neighboring timestamps. The value of μ_t should be provided by user and remains same for all timestamps. α which is predefined parameter reveals users control on this smoothness parameter. The value of α should range from 0 to 1.

In Eq. 4 the numerator denotes the strength of community z at the timestamp t and the denominator characterizes the complete community strength through all the communities at the timestamp t [2]. PACS algorithm is given as follows:

Algorithm 1- PACS Algorithm
Input- Series of directed networks $(1 \leq t \leq T)$, μ_t – Time based smoothness parameter, ^C- Community Group matrix and α.
Output - Community Strength matrix A of size K×T.

1. set t to 1
2. start
3. Find out communities C_t w.r.t. each timestamp;
4. Generate the community group ^C;

```
repeat
    Calculate $a_{zt}$ using  Eq.4;
    Set t to t+1
until t >t
Output A
End
```

C. Measuring change in community Strength using Community Strength Progression Net-

A two-part network that characterizes the relationship between communities detected at timestamp t-1 and communities detected at timestamp t is built. In this network, the nodes on the left denote the communities identified at preceding timestamp, the nodes on the right denote the communities identified at the present timestamp and the edges joining the nodes denote the impact transmission between the communities. Community Progression net is a flow of Strength transmission from C_i^{t-1} toC_i^t. $a_{it}$is the strength of community
i at time t and p_{ij} is the relationship between community i and j, $a_{it}p_{ij}$ can reveal the impact community j gained from community i. The network replicating this transmission relationship is called as **Strength Transmission Net**. Similarly, the strength that the current community j receives from community i is defined as $p_{ij}a_{it}$, which is called as **Strength Reception Net.**

The relationship matrix P of size $K_{t-1} \times K_t$ that represents the relationships between communities taken at neighboring timestamps (t-1 and t) can be calculated as follows:

$$P = {}^{\wedge}D^{-1}\, S^{t-1} C^{t-1} C^{t^{Trans}} S^{t^{Trans}} \tag{5}$$

where ^D is a diagonal matrix and it is used for normalization. ${}^{\wedge}D_{ii} = \sum_{i=1}^{K_t} S^{t-1} C^{t-1}\, C^{t^{Trans}}\, S^{t^{Trans}}$. Using S^t and C^t the value of P can give us relationship with common members and underlying relationships between two timestamp's communities.

D. Finding Top K Strongest Communities-

By using output of Algorithm 1 one can calculate an overall strength for each community, which is beneficial to detect communities that are the strongest/weakest during the complete observation period. There are two approaches to summarize the time-based community strength marks:

1. Unweighted – Here each time-based mark has to be of equal position and can be calculated as follows:

$$\frac{\sum_{t=1}^{T} a_{zt}}{C_z}$$

2. Weighted - However, in this case, the community strength is more significant at some specific timestamps because of which one should give different weights to different timestamps and given as: $\frac{\sum_{t=1}^{T} l^t a_{zt}}{c_z}$, where l^t is the weight for the particular timestamp t. Note that in both cases size of community z.

For testing purpose we have taken network of synthetic dataset. All nodes in this network are connected with edges to each other. Two snapshots of same network are considered for experimental purpose and both undirected as well as directed networks are considered. Fig.3 and Fig.4 shows the comparison between the community strengths that we have achieved from undirected network and directed network. From this figure we conclude that because of link directions the community strength gets increased.

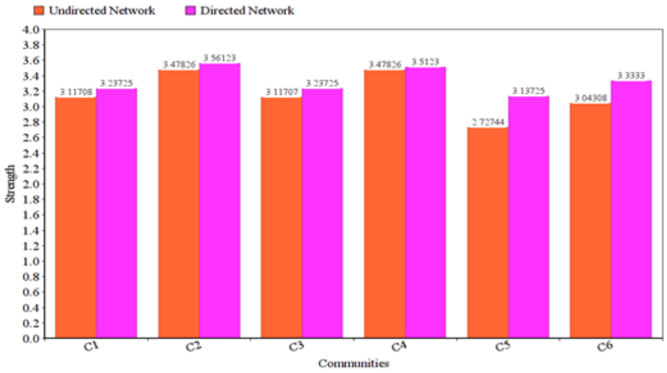

Fig.3 Comparison between Existing System and Proposed System

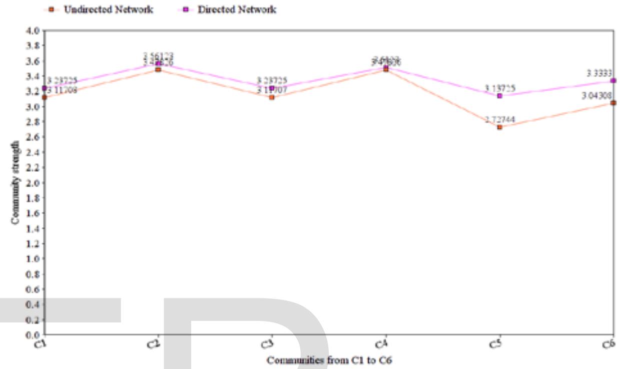

Fig.4 Comparison between Undirected Network and Directed network

The algorithm PACS is compared with 3 other existing methods like PACSwithout, KNN (K- Nearest Neighbor) and CID (Community Internal Density). Fig. 10 shows the global performance on dataset. PACS method gives higher rank than other three methods. PACSwithout uses all the steps in our proposed method except the treatment of the smoothness parameter. Comparison with this PACSwithout will prove the significance of the smoothness notion. The PACS algorithm gives higher performance for both directed networks and undirected networks which is higher than all other existing systems.

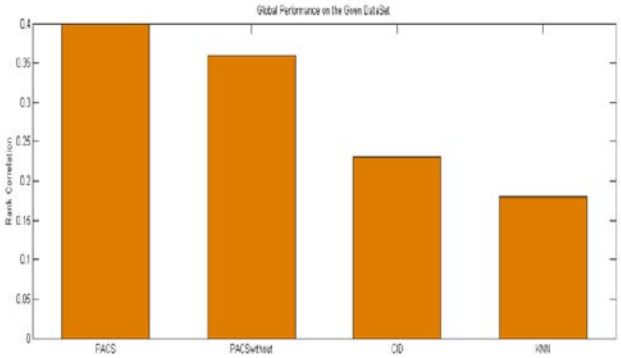

Fig.5 Comparison between Existing algorithms and Proposed algorithm

4 Conclusion and Future Work

A two-stage framework for community detection using NMF technique and community strength analysis using PACS algorithm in directed networks is proposed. The outcomes of community strength analysis are useful to find the top-K strongest or weakest communities and track the change of strengths. This is helpful to military and forensic detectives to intensely recognize hierarchies within criminal organizations.

As a part of future work, content information can be included into our framework not only link information. Also some method for automatically selecting the number of communities can be incorporated in this framework.

References

[1] R. Ahmed and G. Karypis, "Algorithms for mining the evolution of conserved relational states in dynamic networks", *Knowledge and Information Systems*, vol. 33, no. 3, pp. 603-630, 2012.

[2] N. Du, J. Gao and A. Zhang, "Progression analysis of community strengths in dynamic networks",*Prof. IEEE transactions on knowledge and data engineering*,, vol. 27, no. 11, 2015.

[3] P. Bródka, S. Saganowski and P. Kazienko, "GED: the method for group evolution discovery in social networks", *Soc. Netw. Anal. Min.*, vol. 3, no. 1, pp. 1-14, 2012.

[4] T. Falkowski, J. Bartelheimer and M. Spiliopoulou, "Mining and visualizing the evolution of subgroups in social networks.", *Proc. IEEE/WIC/ACM International Conference on Web Intelligence*, 2006.

[5] M. Gupta, J. Gao, Y. Sun and J. Han, "Integrating community matching and outlier detection for mining evolutionary community outliers", *Prof. of KDD'12*, 2012.

[6] Y. Lin, Y. Chi, S. Zhu, H. Sundaram and B. Tseng, "Analyzing communities and their evolutions in dynamic social networks", *ACM Trans. Knowl. Discov. Data*, vol. 3, no. 2, pp. 1-31, 2009.

[7] M. Newman and M. Girvan, "Finding and evaluating community structure in networks", *Physical Review E*, vol. 69, no. 2, 2004.

[8] F. Wang, T. Li, X. Wang, S. Zhu and C. Ding, "Community discovery using nonnegative matrix factorization", *Data Mining and Knowledge Discovery*, vol. 22, no. 3, pp. 493-521, 2010.

[9] M. Kolar, L. Song, A. Ahmed and E. Xing, "Estimating time-varying networks", *Ann. Appl. Stat.*, vol. 4, no. 1, pp. 94-123, 2010.

[10] Y. Park and J. Bader, "How networks change with time", *Bioinformatics*, vol. 28, no. 12, pp. i40-i48, 2012.

[11] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization",*Physical Review E*, vol. 72, no. 2, 2005.

[12] S. Sobolevsky, R. Campari, A. Belyi and C. Ratti, "General optimization technique for high-quality community detection in complex networks", *Physical Review E*, vol. 90, no. 1, 2014.

[13] A. Lancichinetti and S. Fortunato, "Community detection algorithms: A comparative analysis",*Physical Review E*, vol. 80, no. 5, 2009.

[14] S. White and P. Smyth, "A spectral clustering approach to finding communities in graphs", *Prof. SIAM Int.Conf. Data Mining (SDM 05)*, no. 7684, 2005.

[15] B. Rees and K. Gallagher, "Overlapping community detection using a community optimized graph swarm", *Soc. Netw. Anal. Min.*, vol. 2, no. 4, pp. 405-417, 2012.

[16] Y. Chi, X. Song, D. Zhou, H. Hino and B. Tseng, "Evolutionary Spectral Clustering by Incorporating Temporal Smoothness", *KDD'07*, 2007.

[17] L. Tang, H. Liu, J. Zhang and Z. Nazeri, "Community Evolution in Dynamic Multi-Mode Networks", *KDD'08*, 2008.

[18] P. Mucha, T. Richardson, K. Macon, M. Porter and J. Onnela, "Community Structure in Time-Dependent, Multiscale, and Multiplex Networks", *Science*, vol. 328, no. 5980, pp. 876-878, 2010.

[19] K. Xu, M. Kliger and A. Hero, "Tracking Communities in Dynamic Social Networks", *KDD*, 2011.

[20] S. Fortunato, "Community detection in graphs", *Physics Reports*, vol. 486, no. 3-5, pp. 75-174, 2010.

[21] Community Detection in Real Large Directed Weighted Networks", *JDCTA*, vol. 7, no. 5, pp. 521-529, 2013.

[22] M. Newman, "Fast algorithm for detecting community structure in networks", *Physical Review E*, vol. 69, no. 6, 2004.