Tracking Temporal Community Strength in Dynamic Networks

Nan Du, Xiaowei Jia, Jing Gao, Vishrawas Gopalakrishnan, and Aidong Zhang, IEEE Fellow

Abstract—Community formation analysis of dynamic networks has been a hot topic in data mining which has attracted much attention. Recently, there are many studies which focus on discovering communities successively from consecutive snapshots by considering both the current and historical information. However, these methods cannot provide us with much historical or successive information related to the detected communities. Different from previous studies which focus on community detection in dynamic networks, we define a new problem of tracking the progression of the community strength - a novel measure that reflects the community robustness and coherence throughout the entire observation period. To achieve this goal, we propose a novel framework which formulates the problem as an optimization task. The proposed community strength analysis also provides foundation for a wide variety of related applications such as discovering how the strength of each detected community changes over the entire observation period. To demonstrate that the proposed method provides precise and meaningful evolutionary patterns of communities which are not directly obtainable from traditional methods, we perform extensive experimental studies on one synthetic and five real datasets: social evolution, tweeting interaction, actor relationships, bibliography and biological datasets. Experimental results show that the proposed approach is highly effective in discovering the progression of community strengths and detecting interesting communities.

Index Terms—Dynamic Networks, Community Analysis, Community Strength

1 Introduction

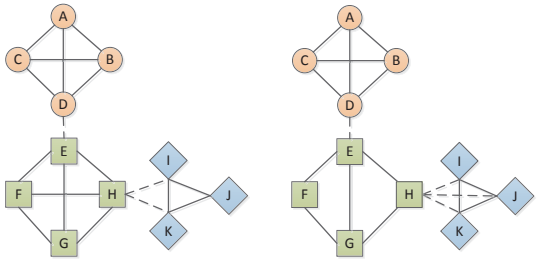
In recent years, there has been a growing interest in modeling and mining various kinds of dynamic networks whose structures evolve over time, such as biological networks, social networks, co-authorship networks and co-starring networks. Specifically, people have investigated community analysis in dynamic networks [1]–[4]. The focus is on detecting communities successively from consecutive snapshots by considering the historical information [5]–[7]. Although these methods can give us quite reasonable and robust communities by considering the temporal smoothness, few historical and successive information related to these communities are provided. Thus we do not know when these communities were assembled or when they are going to disband. Aiming to answer these questions, we propose a novel measure called *community strength*, which can reflect a community's temporal community robustness and coherence throughout the entire observation period.

In this paper, we define that a community is with high strength if it has relatively stronger internal interactions connecting its members than the external interactions with the members to the rest of the world. Dense internal interactions and weak external interactions guarantee that the community is under a low risk of member change (current members leaving or/and new members joining). Intuitively, a friend community is "strong" if its members tie together closely and ignore the temptation from the outside world. On the contrary, a friend community is regarded as a "weak" community if it is likely to confront a member alteration situation. To illustrate this concept, Fig. 1(a) shows a toy example, where the nodes represented by the same geometric shape belong to the same community, solid lines represent internal interactions and dash lines represent external interactions. The circle community (i.e. nodes A, B, C and D) is considered to be stronger than the rectangle community (i.e. nodes E, F, G and H), due to the weaker external attractions. On the other hand, node H has a close relationship with the diamond community (i.e. nodes I, J and K), which makes the rectangle community in the risk of losing its members. In other words, the higher strength score a community obtains, the less possible member alternation occurs in it. It is worth noticing that community strength is a measure which synthetically considers both the community cohesion (i.e. how close the members are in a community) and separation (i.e. how distinct a cluster is from the other clusters).

Furthermore, community strength should be a temporal measure whose value may change as the network evolves. Here's an example in the real world. A set of authors have collaborated closely from 2000 to 2006. During this period, they cooperated frequently among themselves and barely with others outside the community. However, after 2006, because of interest changes, some authors' attentions have been attracted to some other fields. Thus the internal cooperation decreased and the external cooperation increased. In this case, this author community's strength is high and stable during 2000-2006, but begins to decrease after 2006. As a toy example, in Fig. 1(b) (i.e. the network in the $2^{nd} snapshot$

• *All of the authors are with the Computer Science and Engineering Department, State University of New York at Buffalo, Buffalo, NY 14260. E-mail:nandu,xiaoweij,jing,vishrawa,azhang@buffalo.edu*

) which evolves from Fig. 1(a) (i.e. the network at the $1^{st}snapshot$), the strength of the rectangle community decreases, because the internal connections become weaker and external connections become stronger.



(a) $1^{st}snapshot$ (b) $2^{nd}snapshot$

Fig. 1: A Toy Example Illustrating Community Strength

Discovering the progression of community strengths can offer significant insights in a variety of applications. It can help us discover some interesting community information which can not be directly obtained from traditional community analysis. Interesting examples of communities' strength progression can be commonly observed in real-life scenarios. Here we discuss two specific cases in detail.

Strengths Progression in Actor Community: As a strong actor community, the cooperation should be more frequent between the members themselves than between members and non-members. For example, considering the popular and long-running television sitcom 'Friends'[1], its six main actors *J. Aniston*, *C. Cox*, *M. Perry*, *M. LeBlanc*, *L. Kudrow* and *D. Schwimmer* collaborated closely when this sitcom was aired from 1994 to 2004. Let's consider each year's co-starring relationships as one snapshot. We can see that the strength of this community is very low before 1994 (little cooperation between them), and then dramatically increases and keeps stable from 1994 to 2004 (average 23 episodes each year). Finally, the strength of this community apparently becomes weaker after 2004 (much less cooperation comparing to the previous years). The progression of this actor community's strengths shows an interesting pattern of cooperation history among these six actors. Learning the strength progression of actor communities helps us better understand the entertainment industry.

Strength Progression in Gene Community: In the biological domain, the interactions between genes change gradually in dynamic gene co-expression networks. Thus the strength of gene communities also changes. For example, it has been reported that the expression profiling of some key genes will change [8] as the cancer progresses. In such cases, the corresponding gene communities' strength also changes. Discovering the strengths of gene communities throughout a specific disease progression can help us find significant clues in the fields of medicine and biology. For a specific disease, if a gene community is found strong only at the early stage, it is very likely to be a crucial trigger for the disease deterioration.

From the above cases, we can see that discovering the progression of community strengths helps us understand the underlying behavior of communities. The initial idea was published in [9], which covers the basic definition of community strength and the evolutionary analysis on dynamic networks. By utilizing the community strength value, the consistent communities can be detected and tracked over an observation period. This paper extends the original idea to formulate a solid method with broader applications and provide more supportive and comprehensive experiments. In this paper, our goal is to detect the temporal strength of each detected community throughout all the snapshots so that we can answer the following questions: How does the strength of each community change over the observation period? What are the top-K strong communities throughout the observation period? How do the communities from adjacent snapshots influence the strength of each other?

To sum up, our main contributions in this paper are as follows:

- We introduce the notion of *progression analysis of community strengths*. To the best of our knowledge, this is the first work on analyzing the temporal community quality or structure information considering both time and community information.
- We formulate the problem as an optimization framework that can effectively detect the temporal strength of communities and track the strength progression pattern.
- Experiments on the synthetic dataset show the proposed approach is effective on identifying strong communities. On real datasets, interesting and meaningful communities are detected. Case studies suggest that the proposed approach can provide more reasonable results.

The organization of the paper is as follows: In Section 2, we describe the setting of our problem. Section 3 presents the analysis and discussions related to the proposed algorithm. This is followed by discussions on the extensibility of our approach to other applications - Section 4. Extensive experimental studies are reported in Section 5. We then recapitulate related existing work in Section 6 before concluding our work in Section 7.

2 Problem Setting

In this section, we first introduce the definition of community strength and related notations, and then formally define the problem. Before proceeding further, we introduce the notation that will be used in the following discussion: Let a matrix be represented with uppercase letter (e.g. D), d_{ij} denotes the ij-th entry in D, and $d_{i.}$ and $d_{.j}$ denote vectors of i-th row and j-th column of D, respectively. Now, let us start by introducing the definition of the community strength.

1. http://www.imdb.com/title/tt0108778/

Community strength: Given a network $G = (N, E, W)$ where N is the set of nodes in this network, E is the set of edges connecting the nodes, and W is a symmetric weight matrix representing the weights on edges. There have been some existing work on measuring the strength of community by considering its internal compactness or identifying outlier data with probability model [10]. In this paper, we propose the measurement for the community strength which very well fits our problem in real scenarios. The community strength of a community z can be defined as:

$$Strength(z) = \sum_{i\in N}\sum_{j\in N} w_{ij} * \sum_{k\in z}\sum_{l\in z} w_{kl} - \left(\sum_{k\in z}\sum_{v\in N} w_{kv}\right)^2, \quad (1)$$

where $\sum_{i\in N}\sum_{j\in N} w_{ij}$ denotes the sum of all edge weights in the network, $\sum_{k\in z}\sum_{l\in z} w_{kl}$ denotes the sum of internal edge weights inside community z, and $\sum_{k\in z}\sum_{v\in N} w_{kv}$ denotes the sum of internal and external edge weights attached to nodes in community z. The term $\sum_{i\in N}\sum_{j\in N} w_{ij}$ is adopted to guarantee the positive strength value. We propose Eq. 1 inspired by the modularity definition in [11], where the metric is used to measure the quality of the overall network partitioning. The rationale of Eq. 1 is that a strong community should simultaneously obtain dense internal connections and sparse external connections.

Now, our problem can be defined as follows:

Input:

- A series of undirected networks $G^t = (V, E^t, W^t)$ $(1 \le t \le T)$, where each network has N nodes (i.e. $|V| = N$). For each snapshot t, V is a set of nodes, E^t is a set of interactions between these nodes and $W^t_{N\times N}$ is a symmetric weight matrix. For $v_i, v_j \in V$, w^t_{ij} indicates the interaction frequency between nodes v_i and v_j at snapshot t. Note that edges in E^t $(1 \le t \le T)$ could be weighted or unweighted.

Output:

- Community Pool Matrix: We summarize the communities detected from all the snapshots into an $N \times K$ community pool matrix $\tilde{C}$ where K is the number of all the unique communities (i.e. $K = |\tilde{C}|$). In addition, $\tilde{C}$ equals to $C^1 \cup C^2 \cup, ..., \cup C^T$ where C^t $(1 \le t \le T)$ is the temporal community indicator matrix with respect to a certain snapshot t (more details will be introduced later).
- Strength for each community at each snapshot: Let a $K \times T$ matrix A denote the temporal strength for all detected communities, where a_{kt} refers to the strength of community k at snapshot t.

To derive the output, the following variables are needed.

Nuisance Parameters:

- Temporal Community Indicator Matrices: At each snapshot t, the community indicator matrix C^t $(1 \le t \le T)$ is an $N \times K_t$ matrix where K_t is the number of communities captured at snapshot t. If node i is assigned to community k at snapshot t, then $C^t_{ik} = 1$ and 0 otherwise. As we mentioned, all the temporal community indicator matrices C^t $(1 \le t \le T)$ compose the community pool matrix $\tilde{C}$. Note that the communities represented in this matrix can be either overlapping or non-overlapping.
- Temporal Community Relationship Matrices: At each snapshot t, we denote the community relationship matrix S^t as a $K_t \times K_t$ matrix. Note that s^t_{ij} represents the similarity between community i and community j that are detected at snapshot t.

Table 1 summarizes the important notations that we use in this paper.

TABLE 1: Table of Notations

Symbol	Definition
$W^t_{N\times N}$	w^t_{ij}: weight between object i and j at time t
$C^t_{N\times K_t}$	c^t_{ik}: indicator of object i in community k at time t
$\tilde{C}_{N\times K}$	$\tilde{c}_{ik}$: object i grouped into community k
$A_{K\times T}$	a_{kt}: strength score of community k at time t
$1, ..., T$	snapshot indexes
$1, ..., N$	object indexes
$1, ..., K_t$	community indexes at time t
$1, ..., K$	community indexes in the community pool

3 METHODOLOGY

In this section, we present our method for solving the problem of temporal community strength analysis. We begin by introducing the method of partitioning the network from each snapshot into communities in Section 3.1, and then show the method of tracking the strength of each community over time in Section 3.2.

3.1 Community Detection at Each Snapshot

Given a series of temporal networks $G^t = (V, E^t, W^t)$ $(1 \le t \le T)$, we first partition each network independently into K_t communities at each timestamp t. Due to the change of network, the value of K_t may not be the same across different snapshots. Then we store all the detected communities from all the snapshots in a community pool.

To detect communities from each temporal network, we use Non-negative Matrix Factorization (NMF) technique [12]. There are two major reasons to choose NMF: First, it can be easily applied to both hard clustering (i.e. each object belongs to exactly one community) and soft clustering (i.e. each object can belong to multiple communities). The property of soft clustering very well fits many real social scenarios. For instance, each user in social network usually participates in more than one discussion group, as he may have a variety of interested topics. Second, it could uncover the underlying inter-community relationships quite accurately, that can be utilized for other related tasks like progression analysis - refer Section 4.1. The details of these advantages are discussed further in the following discussion of the

method. Please note that we believe one can opt to use other evolutionary clustering algorithm so long as it provides a mechanism for soft clustering and also the ability to identify inter-community relationships.

In this paper, we mainly focus on the undirected network, where the matrix W is symmetric, the clustering to the rows and columns should be identical. Hence we propose to symmetrically factorize each temporal network as follows:

$$\min_{C^t \geq 0, S^t \geq 0} \left\| W^t - C^t S^t C^{t^{Trans}} \right\|^2, \; s.t. \; C^{t^{Trans}} C^t = I. \quad (2)$$

W^t is an $N \times N$ symmetric matrix that demonstrates the interactions between objects at time t. C^t is an $N \times K_t$ community indicator matrix, each entry of which represents the probability of assigning an object into a community. S^t is a $K_t \times K_t$ matrix, providing the relationship between communities detected at time t. Both C^t and S^t should be non-negative. To solve this problem, we propose the following procedure to iteratively update C^t and S^t. Specifically, at iteration t, when S^t is fixed, we update C^t as:

$$c^t_{ij} \leftarrow c^t_{ij} \sqrt{\frac{(W^{t^{Trans}} C^t S^{t^{Trans}})_{ij}}{(C^t C^{t^{Trans}} W^{Trans} C^t S^{t^{Trans}})_{ij}}}. \quad (3)$$

Similarly, fixing C^t, we can obtain the update rule for S^t as:

$$s^t_{ij} \leftarrow s^t_{ij} \sqrt{\frac{(C^{t^{Trans}} W^{t^{Trans}} C^t)_{ij}}{(C^{t^{Trans}} C^t S^t C^{t^{Trans}} C^t)_{ij}}}. \quad (4)$$

During implementation, we add a small value on the denominator to make sure it is not zero. We iteratively update C^t and S^t until convergence. It is worth noticing that the community indicator matrix C^t can be used to derive both hard clustering and soft clustering. Each row of C^t denotes the chance that the corresponding object belongs to the K_t communities. Thus, to get hard clustering results, each object can be assigned to the community with the largest value in the corresponding row of the community indicator matrix. As for soft clustering, we can simply set up a cut-off threshold for the row-based normalized community indicator matrix C^t so that the object is assigned to the cluster whose value in C^t is greater than the threshold.

Moreover, it can be proved that $s^t_{lk} \approx C^{t^{Trans}}_{l.} W^t C^t_{k.} = \frac{1}{|C^t_l||C^t_k|} \sum_{i=1}^{K_t} \sum_{j=1}^{K_t} W^t_{ij}$ [13]. It can be seen that s^t_{lk} demonstrates the relationship between the l-th community and the k-th community. If the network is well-separated, hard clustering can be applied and then S^t is approximately a diagonal matrix where off-diagonal elements are much smaller than diagonal elements. When there exist overlapping between communities, soft clustering is more appropriate, and thus the difference between diagonal and off-diagonal elements is smaller than that observed in the hard clustering case. Therefore, S^t can uncover the underlying relationships between communities more accurately. It is better than the approach that only compares the memberships between communities using measures such as Jaccard coefficient [14]. Furthermore, S^t can be used to construct the strength progression net which will be introduced in Section 4.1. These advantages justify our usage of non-negative matrix factorization to decompose each temporal network.

S^t only captures community relationships at each snapshot, but we are interested in its evolution. Therefore, to consider the communities from different timestamps, we propose to put all the detected communities into a community pool $\tilde{C}_{N \times K}$ and derive the evolutionary pattern. This community pool covers a larger candidate set where all the communities can be compared. Based on this community pool, we plan to find out which communities are grouped closely and consistently over the entire tracking period and which communities are grouped temporarily.

3.2 Temporal Community Strength Analysis

Now, we propose an integrated optimization framework that conducts community strength estimation across snapshots. A naive approach for this task is to calculate the strength of each community individually at each snapshot and track the evolution. However, this approach does not take historical information into account when deriving community strengths and the communities derived across snapshots are not easily comparable. In contrast, we propose the following framework based on the smoothness assumption in which both current and historical networks contribute to the community strength detection. Moreover, in the proposed framework, communities across snapshots are brought into alignment so that we can easily compare them.

Based on Eq. 1, the strength of community z can be further reformulated in terms of the community pool matrix $\tilde{C}$ as follows:

$$\begin{aligned} Strength(z) &= \sum_{i,j=1}^{n} w_{ij} \sum_{i,j=1}^{n} w_{ij} \tilde{c}_{iz} \tilde{c}_{jz} - \left(\sum_{i,j=1}^{n} w_{ij} \tilde{c}_{iz} \right)^2 \\ &= sum(W) \tilde{c}^{Trans}_{z.} W \tilde{c}_{z.} - (\sum_{i=1}^{n} d_i \tilde{c}_{iz})^2 \\ &= \tilde{c}^{Trans}_{z.} (\tilde{W} - D) \tilde{c}_{z.}, \end{aligned} \quad (5)$$

where $sum(W)$ denotes the sum of weights for network W and $\tilde{W}$ equals to $sum(W) * W$. D equals to dd^{Trans} where d is an $N \times 1$ vector such that each d_i is the degree of vertex i. Employing Eq. 5, we formulate the task of estimating a particular community z's strength at snapshot t (referred as a_{zt}) as the following objective function:

$$\min_{a_{.t}} J(a_{.t}) = \alpha \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.}\right] + (1-\alpha) \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right] \quad s.t. \quad \sum_{z=1}^{K} a_{zt} \leq \mu_t, \ a_{zt} \geq 0, \tag{6}$$

where $\log(\frac{1}{a_{zt}})$ determines the z-th community's strength weight corresponding to the snapshot t and μ_t is the estimated sum of community strengths with respect to the current snapshot. For the sake of simplicity, we set μ_t as 1 in the experiments. We propose the objective function Eq. 6 based on Eq. 1 to combine the historical information and the current snapshot. The temporal community strength is derived from the optimization framework with the assumption of smoothness. In real scenarios, the network is usually expected to evolve gradually rather than abruptly. Hence we include the second term in Eq. 6 to combine the strength at previous timestamp. $\log(\frac{1}{a_{zt}})$ is used to capture community strength weight due to the following reasons. First, the logarithm function helps rescaling the strength values by converting them into a small range. Second, it makes the optimization function easier to solve as negative logarithm is a convex function. The stronger a community z at snapshot t, the higher a_{zt} will be and hence $\log(\frac{1}{a_{zt}})$ will be lower. Therefore, to optimize the function, the higher weight will be associated with the community that is stronger at the current snapshot.

Importantly, smoothness is considered in the objective function. Note that in Eq. 6, the first term $\alpha \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.}\right]$ measures the cost of all the detected communities in the community pool with respect to the current snapshot's network, where a high cost means the communities in this snapshot are weak. The second term $(1-\alpha) \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right]$ denotes the temporal smoothness in terms of the goodness of the current clustering result with respect to the previous network, where a higher temporal cost means that the smoothness assumption is violated and inconsistency is observed across snapshots. Therefore, this objective function can better capture the strength calculation at each snapshot and across snapshots.

Temporal Smoothness

In many real-world dynamic network applications, networks are expected to change gradually and stably. Examples include geometric networks [15] and gene networks [16]. As a consequence, we expect a certain level of temporal smoothness between community strengths in successive snapshots. The temporal community strength should depend on the current network, and it should not deviate too dramatically from the previous snapshot's network. Actually, temporal smoothness assumption has been adopted in many previous evolutionary clustering work [6], [7], [17]. However, instead of applying the smoothness among the clusters detected in adjacent timestamps as previous work did, we have applied it on the temporal community strength. In Eq. 6, the overall cost of the objective function is represented as the linear combination of the cost of community strength fitting to the current snapshot and the cost of community strength fitting to the previous snapshot. Thus α $(0 \leq \alpha \leq 1)$ is a predefined parameter to reflect users' emphasis on the smoothness assumption. Usually, α could be assigned a relatively large value when the networks are stable and evolve slowly (e.g., social networks). α should be assigned a relatively small value when the target networks include noise and are likely to evolve swiftly.

PACS Algorithm Procedure

Now, we derive the solution for the community strength scores a_{zt} for objective function shown in Eq. 6. Using the method of Lagrangian Multipliers, we can rewrite Eq. 6 as follows:

$$\min_{a_{.t}} J(a_{.t}) = \alpha \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.}\right] + (1-\alpha) \sum_{z=1}^{K} \log(\frac{1}{a_{zt}}) \left[\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right] + \gamma \left(\sum_{z=1}^{K} a_{zt} - \mu_t\right), \tag{7}$$

where γ is a Lagrangian multiplier. Taking the partial derivative of Eq. 7 with respect to a_{zt} and setting the derivative to 0, we obtain Eq. 8 and Eq. 9.

$$a_{zt} = \frac{\alpha \tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.} + (1-\alpha)\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}}{\gamma} \tag{8}$$

$$\gamma = \frac{\sum_{z=1}^{K} \left[\alpha \tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.} + (1-\alpha)\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right]}{\mu_t} \tag{9}$$

Plugging Eq. 9 into Eq. 8, we obtain the solution for a_{zt} which is shown as Eq. 10. In this equation, the numerator measures the strength of the community z at the snapshot t and integrates both the current and historical information. The denominator represents the overall strength across all the communities at snapshot t, which serves as a normalization factor. Furthermore, μ_t, as mentioned before, controls the overall community strength at a specific snapshot. The intuition behind Eq. 10 is that the communities which have compact structure at the current snapshot will be assigned a larger strength score; while the community whose structures are loose will receive a lower value. The algorithm is summarized

in Algorithm 1, and we name our method as ***PACS*** (Progression Analysis of Community Strength).

$$a_{zt} = \frac{\left[\alpha \tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.} + (1-\alpha)\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right]\mu_t}{\sum_{z=1}^{K}\left[\alpha \tilde{c}_{z.}^{Trans}(\tilde{W}^t - D^t)\tilde{c}_{z.} + (1-\alpha)\tilde{c}_{z.}^{Trans}(\tilde{W}^{t-1} - D^{t-1})\tilde{c}_{z.}\right]} \quad (10)$$

Algorithm 1 The PACS Algorithm
Input: A series of temporal networks $W_{N\times N}^t$ $(1 \le t \le T)$, a series of estimated sum of community strength for each snapshot μ^t $(1 \le t \le T)$, community pool matrix $\tilde{C}_{K\times N}$ and a temporal smoothness parameter α
Output: Estimated Community Strength matrix $A_{K\times T}$

1: $t \leftarrow 1$;
2: begin
3: Detect the communities C_t with respect to each snapshot;
4: Generate the community pool $\tilde{C}$;
5: **repeat**
6: Estimate $a_{.t}$ using Eq. 10;
7: $t \leftarrow t + 1$;
8: **until** $t > T$
9: Output A;
10: end

In the case with directed networks, the community detection can be implemented with the revised method based on cuts, spectral clustering, random walks or markov chains [18]. Also we can generalize our model to the directed graph with the asymmetric NMF.

4 Extensibility to other applications

By formulating the problem as the task of measuring the community strength over an observation period, we can extend our method to perform some additional tasks. In this section, we explain the extensibility of Algorithm 1 to: (1) Measure the impact and consequently the change in community strength based on immediate preceding timestamps, and (2) Identify top-k strongest and weakest communities.

4.1 Community Strength Progression Net

The output of Algorithm 1 provides information on how all the communities' strength evolve over time. In addition to that, we also want to know how the communities from immediate preceding snapshots (i.e. C_{t-1} and C_t) influence the strength of each other. To illustrate these relationships, we construct a bipartite network that represents the relationship between communities detected at snapshot t-1 and communities detected at snapshot t. In such a network, the nodes on the left represent the communities detected at previous timestamp, the nodes on the right represent the communities detected at the current timestamp and the edges connecting the nodes denote the influence transmission between the communities.

The relationship matrix $P_{K_{t-1}\times K_t}$ that represents the relationships between communities captured at adjacent snapshots (t-1 and t) can be calculated as:

$$P = D^{-1} S^{t-1} C^{t-1} C^{t^T} S^{t^T}, \quad (11)$$

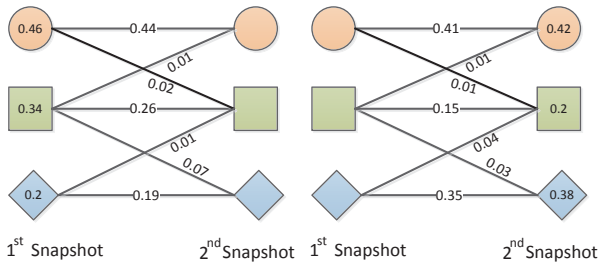
where D is a diagonal matrix used for normalization and $D_{ii} = \sum_{j=1}^{K_t}(S^{t-1}C^{t-1}C^{t^T}S^{t^T})_{ij}$. As we mentioned in Section 3.1, C^t and C^{t-1} are the community indicator matrices with respect to snapshot t and t-1. S^t and S^{t-1} represent the relationship between communities at snapshot t and t-1, respectively. As we mentioned previously, S^t can uncover the underlying relationships between communities detected at snapshot t, and $C^{t-1}C^{t^T}$ demonstrates the number of common members between the two snapshots' communities. Thus, P can reflect not only the common member relationship but also the underlying relationships between two snapshots' communities.

A natural definition of community progression net (from c_i^{t-1} at time t-1 to c_j^t at time t) is a flow starting from c_i^{t-1}, and transmits its strength to c_j^t. There are two applications that are worth discussing: First, we analyze how the community strength from the current snapshot transmits to the next snapshot. Second, we analyze how the current community strength succeeds from the previous snapshot. For the former one, the strength transmits community i at the current snapshot to the community j at the next snapshot, which is defined as $a_{it}p_{ij}$. As mentioned before, a_{it} is the strength of community i at time t and p_{ij} is the relationship among community i and j, $a_{it}p_{ij}$ can reflect the influence community j obtained from community i. The network reflecting this transmission relationship is named *Strength Transmission Net*. Correspondingly, for the latter one, the strength that the current community j inherits from community i is defined as $p_{ij}a_{jt}$, which is named *Strength Reception Net*. Notice that to measure the *Strength Reception Net*, we need to normalize each column of P.

Examples for *Strength Transmission Net* and *Strength Reception Net* are depicted in Fig. 2(a) and 2(b), respectively. In each network, the values shown inside the geometric shapes are the strength corresponding to the communities. For example, from Fig. 2(a) we can see that the circle community from the 1^{st} snapshot transmits its current strength (0.46) to the succeeding circle community with 0.44 and rectangle community with 0.02. Take another example, from Fig. 2(b), we can see the diamond community at the 2^{nd} snapshot inherits 0.35 and 0.03 strength from diamond community and rectangle community at the 1^{st} snapshot. In such cases, we can find out that the members from diamond community at the 2^{nd} snapshot mainly inherits from diamond community at the 1^{st} snapshot.

The community strength progression net can provide us with important information about the community evolution. For example, in the field of biology, if one gene community's strength mainly transmits to multiple subsequent gene communities, it is very likely that this gene community has splitted into these communities. Take another example in the social network, if several

friend communities' strengths have transmitted to one subsequent community, we would know that these previous communities merge into a larger community.



(a) Strength Transmission Net (b) Strength Reception Net

Fig. 2: Strength Progression Nets of the Toy Example.

4.2 Top-K strongest/weakest communities

By applying Algorithm 1, we obtain the community strength for each detected community at each snapshot. Based on this output, we can compute an overall strength for each community, which is useful to identify interesting communities that are the strongest/weakest throughout the entire observation period. There are mainly two methods to aggregate the temporal community strength scores: unweighted and weighted. In the unweighted case, we can regard each temporal score to be of equal importance and take the sum, i.e. $\sum_{t=1}^{T} a_{zt}$. However, in some cases, the community strength is more important at some particular snapshots, e.g. the early stage of cancer. In such a case, we should give different weights to different snapshots and the aggregation function can be defined as $\sum_{t=1}^{T} h^t a_{zt}$, where h^t is the weight for the specific snapshot t. In addition, when choosing the top strongest or weakest communities, we may also want to consider the size of the communities. When the target networks are very sparse, the penalty from the external connections may be very small, thus the penalty from the external interaction would be very limited. In such a case, the community strength value would be biased to the large-size communities which will contain more internal connections. To mitigate this effect, the aggregated function for community z can be redefined as: $\frac{\sum_{t=1}^{T} a_{zt}}{|C_z|}$ or $\frac{\sum_{t=1}^{T} h_t a_{zt}}{|C_z|}$ so that the community strength is normalized by its size.

5 Experiments

In this section, we report experimental studies based on both synthetic and real-world datasets. First, we evaluate the proposed method by comparing the detected strongest/weakest communities and community strength ranking with the ground truth on synthetic and real-world social datasets. Then we evaluate the results obtained by the proposed method on actor relationship, bibliography and biological datasets using case studies. We perform comprehensive analysis to justify the top-K strongest communities returned by the proposed algorithm.

5.1 Synthetic Dataset

We start with a synthetic dataset, which is generated according to the method mentioned in [7]. We generate data for a total of 30 consecutive snapshots. At the 1^{st} snapshot, we generate 100 nodes, which are divided into five communities of 20 nodes each. From the 2^{nd} to the 30^{th} snapshots, edges are added randomly with a higher probability p_{in} for within-community edges and a lower probability probability p_{out} for between-community edges. In this study, we set the two-tuple parameter (p_{in}, p_{out}) for these five communities as $C_1 = (0.22, 0.05)$, $C_2 = (0.2, 0.07)$, $C_3 = (0.18, 0.09)$, $C_4 = (0.16, 0.11)$ and $C_5 = (0.14, 0.13)$.

Baselines

Since no previous methods target at the same problem, we compare the proposed algorithm with several variations of previous approaches and the proposed approach.

$PACS_{without}$: The first baseline, which we call $PACS_{without}$, adopts all the steps in the proposed method except the usage of the smoothness constraint. Comparison with $PACS_{without}$ will demonstrate the importance of the smoothness assumption.

KNN: The second baseline is the K-nearest neighbors (*KNN*) approach. At each snapshot, after we use *KNN* to detect the communities, we also calculate the strengths for them. Then, the strength of each specific community can be calculated via considering its top-K most similar communities' structure information. Specifically, in *KNN*, the formula for strength of community i at time t can be represented as $\sum_{j=1}^{K} \tilde{h_{ij}} \overline{a}_{jt}$. Note that, $\overline{a}_{jt}$ is the strength of community j at time t which is calculated via Eq. 5, and $\tilde{h_{ij}}$ here is the similarity between two communities calculated using Jaccard coefficient and $\sum_{j=1}^{K_t} \tilde{h_{ij}} = 1$. *KNN* represents the perspective of involving only k closest communities in strength computation while the proposed approach conducts a global computation of community strength.

CID: The third baseline is based on community internal density, which we call *CID* for short. As defined in [19], $CID_c^t = \frac{\sum_{i \in c} \sum_{j \in c} w_{ij}^t}{|c|(|c|-1)/2}$ measures the internal edge density of the cluster c. Since *CID* is also proposed to measure the community quality, comparing with *CID* can help us understand which community quality index can better measure the community strength.

Performance Evaluation on Synthetic Dataset

Due to the way we generate the synthetic data, we have known that community C_1 has the largest gap between p_{in} and p_{out}, which makes it the strongest community throughout all the snapshots. Thus, we can directly compare the strongest community discovered by each method with C_1. The higher similarity the detected strongest community to C_1, the more accurate the corresponding method is. In this experiment, we

use *Jaccard coefficient* which is defined as the size of the intersection members divided by the size of the union of the members to measure the community similarity. On the other hand, since all these five communities in the synthetic data are more or less well separated, the weakest community should be the one composed of members coming uniformly from these five communities. To measure the distribution of members, we use entropy measure $-\sum_{i=1}^{5} P(x_i)logP(x_i)$, where $P(x_i)$ is the percentage of members from community i. The higher the entropy is, the weaker the discovered community is.

Results on the Synthetic Dataset

Table 2 shows both the Jaccard coefficient for the strongest community and the entropy for the weakest community obtained by the four algorithms. From the table we can see that the proposed method clearly outperforms the baselines on both strong and weak community detection. $PACS$ performs better than $PACS_{without}$, which indicates that the smoothness assumption contributes to the performance improvement. Moreover, *CID* does not perform well, since it only considers the internal connections of the communities.

TABLE 2: Performance Comparison with Baselines

Method	Jaccard coefficient (Strong Community)	Entropy (Weak Community)
$PACS$	**0.95**	**1.38**
$PACS_{without}$	0.90	1.33
KNN	0.85	1.1
CID	0.85	1.33

5.2 Social Evolution Dataset

The social evolution dataset was collected by MIT human dynamics lab [20], which recorded the daily living of 80 students in a dormitory with mobile phones. We construct the student interaction networks from the raw data. In these networks, nodes are students and an edge exists between them if the corresponding students have interactions in one of the following three ways: call, message and music sharing. The weight of each edge is the number of interactions. It is easy to see that, the more frequently two students interact with each other, the higher weight is assigned on the edge connecting them. In addition, there are five snapshots which correspond to the time *before 10/19/2008 - 10/19/2008* (T1), *10/20/2008 - 12/13/2008* (T2), *12/14/2008 - 3/5/2009* (T3), *3/6/2009 - 4/17/2009* (T4) and *4/18/2009 - 5/22/2009* (T5). We will discuss why the time intervals are divided in this way later.

Note that the number of communities at time t can be determined by the *modularity function* [11]. To determine the best community number K at time t, we tried different candidates for K in a specific range and used the one which leads to the highest modularity function value. From these student interaction networks, we hope to rank friend communities based on their strengths. In other words, we try to find out strong friend clans from their regular social interactions.

The evaluation of temporal community strength is difficult due to the lack of ground truth. Fortunately, besides the student interaction information, this dataset also provides a series of temporal surveys about the closeness degree between students, which can be used to validate our discoveries. The surveys were mainly made on five dates: *10/19/2008*, *12/13/2008*, *3/5/2009*, *4/17/2009* and *5/22/2009*. (This is the reason why we cut the snapshots of student interaction networks in the way mentioned above). In each survey, every student needs to indicate his/her current relationship level with the others in the six kinds of surveyed relationships, which are sorted and weighted by us in view of the closeness degree in Table 3.

TABLE 3: Weights for Various Closeness Categories

Original	Redefined	Weight
Close Friend	Friend	3
Socialize Twice Per Week Political Discuss	Acquaintance	2
Facebook All Tagged Photos	Not familiar	1
Blank	Do not know	0

Performance Evaluation on Social Evolution Dataset

We first calculate the gap between the average inside-community closeness degree and the average outside-community closeness degree for each community in the community pool $\tilde{C}$, and then we sort these communities based on their gaps as a ranking list L_1. Note that the higher a community is ranked, the stronger it is. It is obvious that a strong friend clan should simultaneously obtain close relationships among the members and obtain a relatively weaker relationship with non-members. The dense internal connections and sparse external connections situation ensures a low probability of current members leaving and new members joining, which makes it a strong friend clan. Also, we calculate the community strength for each community in the community pool $\tilde{C}$ using the proposed method or one of the baselines, and then also rank them as a ranking list L_2. Then we can directly compare the predicted result L_2 with L_1. Because L_1 denotes the ground truth of the communities' strengths ranking, the estimated result can be validated through the comparison of two ranking lists.

The similarity between ranked list L_1 and L_2 can be measured globally or locally. The global approach measures how close the overall similarity is between L_1 and L_2. To measure this, we use rank correlation measure proposed in [21], which is also commonly referred to as Kendall's tau (τ) coefficient. The Kendall's tau coefficient ranges in [-1,1]. Using this measure, two identical rankings will receive value 1, and the opposite rankings (i.e., one ranking is the reverse of the other) has value -1. Besides measuring the proposed method from the global perspective, we are also interested in the

method's local precision at the two extremes (strongest and weakest) in a ranking list. To measure the accuracy of detected communities in the top/bottom x elements, we use a cover rate function of $\frac{|L_1^{T_x} \cap L_2^{T_x}| + |L_1^{B_x} \cap L_2^{B_x}|}{|L_1^{T_x}| + |L_1^{B_x}|}$, where L^{T_x} denotes the elements of the top x ratio of list L and L^{B_x} denotes the elements of the bottom x ratio of list L. This function is used to measure the common elements ratio of two ranking lists in terms of their $2x$ (top x and bottom x percent) percent elements. In the experiment, we vary the ratio x as 10%, 15%, 20%, 25% and 30%.

The experimental results of global and local ranking measurement comparing with baselines mentioned in Section 5.1 are shown in Fig. 3 and Fig. 4, respectively. It can be observed from these figures that the proposed method outperforms other baselines consistently in both the global measure and local measure. To be more specific, for the global measure, we got the similar performance results as those on synthetic dataset. For the local measure, two patterns are found: for the $PACS$ and $PACS_{without}$, the cover rates are high at the beginning, and begin to decrease as x increases; for the *KNN* and CID, they start with low cover rates and the cover rates increase as the x increases. The reasons behind these patterns are as follows. First, the farther communities ranked from two extremes (i.e. strongest and weakest), the differences between them are more trivial. In other words, the communities which are closer to the middle are hard to rank. This is the reason why the cover rates of $PACS$ and $PACS_{without}$ drop when x increases. Second, since the cover rate measures the two ranking lists' common elements ratio in terms of their $2x$ percent elements, even two ranked lists are totally opposite, they will reach 100% cover rate when x equals to 50%. Also, different with $PACS$ and $PACS_{without}$ whose beginning cover rate is very high, the beginning cover rate of CID and *KNN* is very low, thus the possible growing space of cover rate is larger.

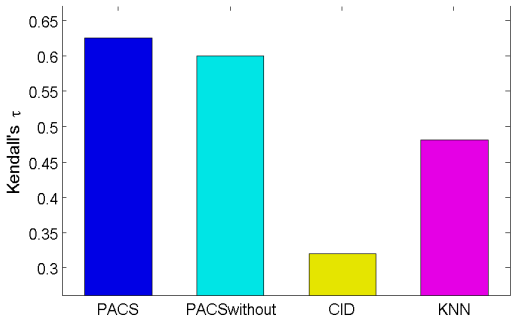


Fig. 3: Global Performance on the Social Evolution Dataset

To evaluate how the temporal smoothness parameter α affects the performance, we increase α from 0 to 1 with a step of 0.1 and report the Kendall's tau value of the detected community ranked list in Fig. 5. As α increases, we emphasize more on the current network. We get a hill-shape curve, which demonstrates that both historical and current networks contribute to the true community strength estimation. In other words, the temporal smoothness assumption adopted in our framework is helpful. On the other hand, we validate the influence from different number of communities in Fig. 6. For simplicity, we set the K_t value identical for each timestamp. In the test, different K_t values are tested and the Kendall's tau coefficients are recorded. It can be observed that the different number of communities will not cast much impact on the efficacy of our framework.

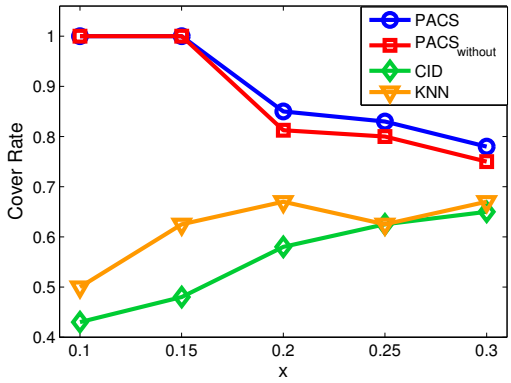


Fig. 4: Local Performance on the Social Evolution Dataset

Finally, we report the average inside-community closeness degree (as mentioned in Table 3) and the average outside-community closeness degree at each snapshot for the top three strongest and weakest communities in Table 4. We can observe that, for the top three strongest communities, the average inside-community closeness degrees are obviously larger than the average outside-community closeness degrees. On the contrary, the average inside-community closeness degrees of weak communities are basically less than the average outside-community closeness degrees. This demonstrates that the temporal community strength detected by the proposed method can reflect the true community relationships.

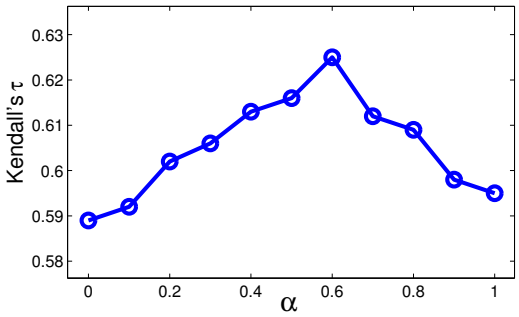


Fig. 5: Parameter Sensitivity

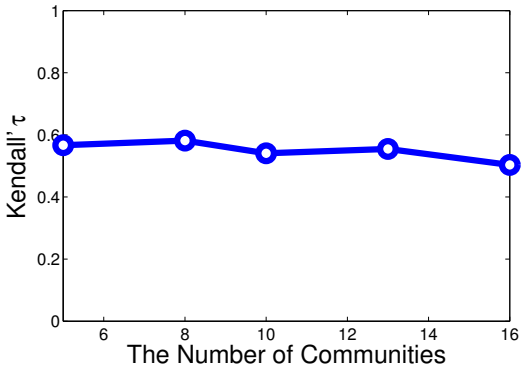


Fig. 6: The Performance on Different Number of Communities

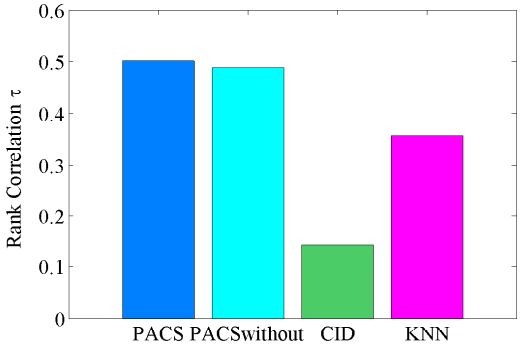


Fig. 7: Global Performance on the Twitter Dataset

	T1	T2	T3	T4	T5
	In/Out	In/Out	In/Out	In/Out	In/Out
Top three strongest communities					
1st	2.1/1.2	2.1/1.2	2.1/1.3	2.0/1.3	2.1/1.3
2st	1.8/1.2	1.8/1.2	2.0/1.3	1.9/1.3	2.1/1.3
3st	2.0/1.2	2.0/1.2	2.0/1.3	1.9/1.3	2.1/1.4
Top three weakest communities					
1st	0.8/1.0	0.7/1.0	0.7/0.9	0.7/0.9	0.6/0.9
2st	0.1/1.1	0.1/1.0	0.7/1.1	0.9/1.2	0.7/1.2
3st	1.5/1.6	2.0/1.5	1.3/1.3	1.1/.2	0.1/1.0

TABLE 4: Inside/Outside Closeness Degrees of the Top Three Strongest/Weakest Communities

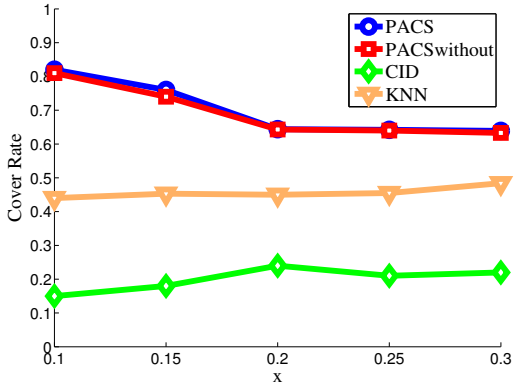


Fig. 8: Local Performance on the Twitter Dataset

5.3 Twitter Dataset

The Twitter Dataset is crawled from Twitter.com[2]. We tracked the topic "English Premier League" and related football teams for over a week and then divide the data into 5 intervals, each spanning 60 hours of interactions. There are in total 1582 users involved in the networks. The weight on the edge stands for the number of interactions between two users in the specific timestamp. To validate our results, we create a word frequency vector for each user and calculate the cosine similarity between a given pair of users. Thus we create a text based similarity matrix as ground truth. Similar with Social Evolution Dataset, we implement our proposed method on user interaction network, and make comparison with the ground truth information.

We conducted the experiments to display the global and local performance, shown in Fig. 7 and Fig. 8 respectively. The stronger community should have discussions on relatively consistent topics, and thus more similar word patterns over time. In the results, our proposed method outpaces the two baseline in Twitter dataset. Also we can observe the similar local performance patterns with that in Social Evolution Dataset from Fig. 8.

5.4 IMDB Dataset

In this part, we consider the co-starring network constructed from a subset of the Internet Movie Database (IMDB) dataset[3]. In the network, nodes are actors and an edge exists between them if the corresponding actors have participated in an American made comedy movie (excluding TV movies and TV series) together in a given period of time. There are totally four snapshots from 1991 to 2002, and the network of each snapshot demonstrates co-starring relations over a period of three years *1991-1993* (T1), *1994-1996* (T2), *1997-1999* (T3) and *2000-2002* (T4). To remove noise and outliers, only actors who have participated in at least 10 movies of any genres from 1990 to 2010, and at least one American made comedy movie are selected at each snapshot. There are totally 700 actors satisfying the above requirements, and we build snapshot networks with these 700 actors as nodes.

Top three strongest actor communities detected by *PACS* and the major movies of each community are shown in Table 5. Without loss of generality, we demonstrate the case study on the first community and show that the method is effective in discovering strong communities. Among all the 18 actors that are included in this community, 12 are indicated as *voice actors* in Wikipedia, and the rest of them have also contributed their voices to some cartoon characters at least three times. Moreover, in their nine key movies, seven are

2. http://www.twitter.com/

3. www.imdb.com/interfaces

comedy cartoons. As we know, the particularity of dubbing makes the cooperation between voice actors more frequent than with the actors who are not voice actors, which demonstrates that this actor community detected by the proposed method is indeed a strong community.

5.5 DBLP Dataset

We evaluate the proposed method on a subset of DBLP Dataset used in [22]. To be more specific, we focus on work published on conferences or journals during 1991-2000 with 144,179 papers in total. We track the strength of the author communities within five time intervals: 1991-1992(T1), 1993-1994 (T2), 1995-1996 (T3), 1997-1998 (T4) and 1999-2000 (T5). Only authors who had at least one publication (in a selected set of 43 conferences/journals) at each timestamp are considered. There are in total 1059 authors who are represented as nodes in the networks. Each node pair is connected if the corresponding authors have joint publications, and the weight connecting the nodes denotes the times of collaboration at this timestamp.

The strongest author community detected by *PACS* and its related collaboration venues within five timestamps are shown in Table 6. This author community consists of 14 authors, and due to space limit, we only provide their abbreviated names. We show the number of collaborations among the authors for each conference/journal they co-published in. As for the top five frequent venues in which they co-published in, on average they have around four co-authored papers in every two years. Furthermore, their publications on these journals/conferences are relatively consistent throughout the observation period. The high frequency and stability of collaboration has made this author community a strong community. Therefore it demonstrates that the strong author community detected by the proposed method is reasonable.

5.6 Biological Dataset

The biological dataset used in the experiment is from *Stevenson et al.* [23]. They collected the gene expression data from two groups of rats: the rats (eight replicates) exposed to cigarette smoke (i.e. exposure group) and the rats (eight replicates) exposed to room air only (i.e. the control group). Various intervals up to eight months are used to identify the molecular changes induced by cigarette smoke inhalation that may drive the biological and pathological consequences leading to diseases, such as asthma and lung cancer. The dataset includes eleven snapshots (1, 3, 5, 14, 21, 28, 42, 56, 84, 112 and 182 days), and there is a gene expression matrix created for each snapshot. In this study, we focus on 3672 genes whose *p-values* (via t-test) are smaller than *0.05*. To construct the gene co-expression networks, we first calculate the *Pearson correlation coefficient* between the gene pairs based on each snapshot's gene expression matrix and then maintain the edges whose correlation coefficients are larger than a cutoff threshold (which is set to 0.8). Note that this is a commonly used way to construct gene co-expression network as used in many previous work [24], [25].

A widely used method to analyze gene clusters is to divide them into functional categories for biological interpretation. This is usually accomplished using Gene Ontology (GO) categories [26]. The GO provides biologists a list of gene annotations which are used as inferences for understanding the genes communities' biological functions instead of investigating each gene individually. When GO is used on the strong gene communities detected from the exposure group of rats, we can find the strongest and most significant gene functions that influence this group of rats throughout the entire observation period. Similarly, we can also obtain the significant gene annotations influencing the control group.

We compare the gene annotations between these two groups and filter the common gene annotations. Then the unique gene annotations influencing the exposure group can be obtained, which can tell us the most significantly affected annotations in the chronic response to cigarette smoke. Table 7 shows all the significant gene annotations in the strongest gene community detected by the proposed approach (for the sake of simplicity, we name this community C^*), under the *p-value* cut-off threshold 2.0E-06. Furthermore, from all these annotations, we select the gene annotations which are only detected in C^* but not detected by the top-10 strongest communities from control group, which are shown in Table 7 with a star mark (*). Among all these unique annotations, majority of them have been proven by previous studies to be really driven by the cigarette smoke. As shown in [23], carbon fixation (i.e., GO ID 19752), metabolism (i.e., GO ID 43436, 6082, 42180, 44281, 8152) and inflammation (i.e., GO ID 6954, 2526) are some special functions distinguishing between the exposure group and control group. In addition, the strength progression of this community is shown in Fig. 9. From this plot, we can see that C^* becomes much stronger after the 4-th snapshot (14 days). This is validated by the previous result provided in [23], which demonstrated that carbon fixation, metabolism and inflammation show differences after two weeks. This interesting result shows that the proposed approach is not only effective on detecting the top strongest communities globally, but also effective on tracking the progression of community strengths locally. Besides the proven gene annotations, we believe that the rest of the unique annotations (i.e. those not proven by previous work) may provide important hints for learning the effect of smoking cigarettes.

6 Related Work

There have been extensive research studies on community detection in networks. [27] came up with an efficient algorithm to conduct overlapping community detection

TABLE 5: Members and Corresponding Major Movies in the Top Three Strongest Actor Communities during 1991-2002

Actors	Key work
Debi Derryberry, Christopher McDonald Erik von Detten, Bob Bergen, Phil Proctor, Sherry Lynn Jim Varney, Mickie McGowan, Tom Hanks, Jack Angel Wallace Shawn, R. Lee Ermey, Harry Shearer, John Mahoney Earl Boen, Ben Stein, Charlton Heston, Wayne Knight	An American Tail: Fievel Goes West (1991) Aladdin (1992), Toy Story (1995) House Arrest (1996), A Smile Like Yours (1997) A Bug's Life (1998), Toy Story 2 (1999) The Iron Giant (1999), Recess: School's Out (2001)
Patrick Richwood, Kathleen Marshall, Garry Marshall Larry Miller, Hope Alexander-Willis, Hector Elizondo Marvin Braverman, Rosie O'Donnell, Shannon Wilcox Sean O'Bryan, Donal Logue, Greg Lewis, Jane Morris	Frankie and Johnny (1991) A League of Their Own (1992), Exit to Eden (1994) Dear God (1996), The Other Sister (1999) Runaway Bride (1999), The Princess Diaries (2001)
John Cusack, Kathleen Doyle, Peter McRobbie Woody Allen, Tony Sirico, Michael Rapaport Paul Herman, David Ogden Stiers Jeff Mazzola, Brian McConnachie John Doumanian, Colicchio Victor Natasha Lyonne, Jack Warden, Alan Alda Alan Alda, Steven Randazzo, Paul Giamatti	Shadows and Fog (1991) Manhattan Murder Mystery (1993) Bullets Over Broadway (1994), Mighty Aphrodite (1995) Everyone Says I Love You (1996) Deconstructing Harry (1997), Celebrity (1998) Small Time Crooks (2000) The Curse of the Jade Scorpion (2001)

TABLE 6: Authors and Corresponding Major Publications in the Strongest Co-Author Communities during 1990-2000

Authors	1991-1992	1993-1994	1995-1996	1997-1998	1999-2000
S. Suri, M. Sharir, Da. Dobkin L. Guibas, J. Snoeyink J. Hershberger, P. Agarwal M. Grigni, B. Chazelle, M. Berg D. Halperin, M. Overmars B. Aronov, D. Kirkpatrick	D. Geometr (2) Comput. Geo. (7) SICOMP (1) SWAT (3) J. Algorithms (3) Others (17)	D. Geometr (8) Comput. Geo. (2) SICOMP (4) SWAT (2) J. Algorithms (4) Others (20)	D. Geometr (9) Comput. Geo. (3) SICOMP (2) SWAT (0) J. Algorithms (4) Others (11)	D. Geometr (4) Comput. Geo. (11) SICOMP (7) SWAT (2) J. Algorithms (0) Others (13)	D. Geometr (8) Comput. Geo. (1) SICOMP (2) SWAT (2) J. Algorithms (1) Others (5)

TABLE 7: Gene Annotations for C^*

GO-ID	*p-value*	Description
10038	2.61E-10	response to metal ion
10035	1.33E-09	response to inorganic substance
19752*	2.21E-08	carboxylic acid metabolic process
43436*	2.21E-08	oxoacid metabolic process
6082*	2.47E-08	organic acid metabolic process
42180*	2.96E-08	cellular ketone metabolic process
9719	1.50E-07	response to endogenous stimulus
44281*	2.77E-07	small molecule metabolic process
10033	3.31E-07	response to organic substance
9725	3.38E-07	response to hormone stimulus
71396*	4.24E-07	cellular response to lipid
2526*	5.01E-07	acute inflammatory response
44283*	5.18E-07	small molecule biosynthetic process
6954*	8.71E-07	inflammatory response
8152*	1.69E-06	metabolic process
6952*	1.92E-06	defense response

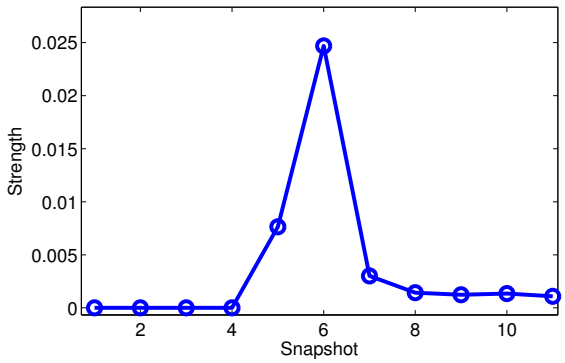


Fig. 9: Community Strength Progression of Community C^*

in large-scale social networks. In [28], a novel method was proposed for the community discovery in complex networks based on an extremal modular optimization framework. [29] introduced the modularity concept in social networks and leveraged eigenvectors of characteristic matrix for the detection task. Also, [30] discussed a benchmark method to test the detected communities. However, these methods focus on the static scenario and cannot be easily extended to dynamic networks.

With the availability of many online datasets, dynamic network analysis has become a hotly discussed topic today. In [1], an optimization framework based on logistic regression was proposed to estimate the network evolution. [2] discussed the biological dynamic networks and proposed the method to predict the state of protein complexes. [3] analyzed the dynamic molecular interactions, which is crucial in regulating the functioning of cells and organisms. In [31], the authors investigated the subgraph discovery in dynamic networks.

Besides, the community analysis of dynamic networks has been extensively studied in various research areas. Most existing community research on dynamic networks focuses on community discovery and community evolution pattern detection. *Chi et al.* [6] and *Lin et al.* [7] proposed community discovery algorithms for dynamic graphs where the communities detected at each snapshot

are based on the optimization defined on both the current and historical networks. Similarly, *Ahmed et al.* [32] proposed a machine learning algorithm named TESLA for recovering the underlying structure of time-varying networks, which could be also used on temporal community detection. Although these methods can output some stable and consistent communities at each snapshot, they may not be able to provide any evolutionary information of the communities, such as the historical/successive structure information. Some studies [33], [34] focused on detecting the evolution of communities, which captures the changes (e.g. merging, splitting and surviving) between successive communities. However, the information provided by these studies is limited to only adjacent snapshots which cannot give us a whole picture of the community evolution.

Gupta et al. [5] investigated and tackled the problem of identifying evolutionary community outliers given the discovered communities from two snapshots of an evolving dataset. The target problem is to detect community outliers that evolve against the trend, which is different from ours. *Newman and Girvan* [11] proposed the *Modularity function*, which measures the quality of graph partitioning. In particular, a graph is considered to have high modularity if it has dense connections between the nodes within the communities but sparse connections between nodes across different communities. *Modularity function* captures community strength in some sense, but it is only used as a global index which measures the quality of a particular clustering or the standard to partition the objects into communities [35] in a static graph.

7 Conclusions

In this paper, we introduced a new problem of analyzing the progression of community strengths. Community strength is a temporal measure which represents the probability that a particular community has a stable membership at the current snapshot. To solve this problem, we propose a framework that provides reliable and consistent community strength scores which are not only insensitive to short-term noise in the current network but also adaptive to long-term network evolution. The results of community strength analysis can be also used to find the top-K strongest or weakest communities and track the change of strengths via constructing the community strength progression net. Extensive experimental analysis demonstrated that the proposed method is very effective on both synthetic and real dynamic datasets. Case studies on three real datasets showed that interesting and meaningful communities can be revealed by community strength detection.

8 Acknowledgement

This work was supported in part by the National Science Foundation under Grant NSF IIS-1218393, 1016929 and 1319973. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] M. Kolar, L. Song, A. Ahmed, and E. P. Xing, "Estimating time-varying networks," *ArXiv e-prints*, 2008.

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

[3] T. M. Przytycka, M. Singh, and D. K. Slonim, "Toward the dynamic interactome: it's about time," *Briefings in Bioinformatics*, vol. 11, no. 1, pp. 15–29, 2010.

[4] P. Bogdanov, M. Mongiovi, and A. Singh, "Mining heavy subgraphs in time-evolving networks," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, 2011, pp. 81–90.

[5] M. Gupta, J. Gao, Y. Sun, and J. Han, "Integrating community matching and outlier detection for mining evolutionary community outliers," in *In Prof. of KDD'12*, 2012, pp. 859–867.

[6] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng, "On evolutionary spectral clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 4, pp. 1–30, 2009.

[7] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng, "Analyzing communities and their evolutions in dynamic social networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 2, pp. 1–31, 2009.

[8] A. L. Creekmore, W. T. Silkworth, and et al., "Changes in gene expression and cellular architecture in an ovarian cancer progression model," *PLoS ONE*, vol. 6, no. 3, pp. 1–16, 2011.

[9] N. Du, J. Gao, and A. Zhang, "Progression analysis of community strengths in dynamic networks," in *Prof. of ICDM'13*, 2013.

[10] L. J. Deborah, R. Baskaran, and A. Kannan, "A survey on internal validity measure for cluster validation," *International Journal of Computer Science & Engineering Survey*, vol. 1, no. 2, pp. 85–102, 2010.

[11] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E - Statistical, Nonlinear and Soft Matter Physics*, vol. 69, no. 2 Pt 2, pp. 1–16, 2003.

[12] P. Paatero and U. Tapper, "Positive matrix factorization: A nonnegative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[13] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal nonnegative matrix t-factorizations for clustering," *In Prof. of KDD'06*, pp. 126–135, 2006.

[14] P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et des Jura," *Bulletin del la Société Vaudoise des Sciences Naturelles*, vol. 37, pp. 547–579, 1901.

[15] L. Guan and M. Duckham, "Decentralized reasoning about gradual changes of topological relationships between continuously evolving regions," in *Spatial Information Theory*, 2011, vol. 6899, pp. 126–147.

[16] S. Wu and X. Gu, "Gene network: Model, dynamics and simulation," *Computing and Combinatorics*, vol. 3595, pp. 12–21, 2005.

[17] M.-S. Kim and J. Han, "A particle-and-density based evolutionary clustering method for dynamic networks," *In Proc. VLDB Endow.*, vol. 2, no. 1, pp. 622–633, 2009.

[18] S. E. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007.

[19] J. Leskovec, K. J. Lang, and M. W. Mahoney, "Empirical comparison of algorithms for network community detection," *In Prof. of WWW'10*, 2010.

[20] A. Madan, M. Cebrian, S. Moturu, K. Farrahi, and A. S. Pentland, "Sensing the 'health state' of a community," *IEEE Pervasive Computing*, vol. 11, no. 4, pp. 36–45, 2012.

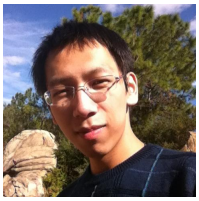
[21] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1/2, pp. 81–83, 2010.

[22] L. Tang, H. Liu, J. Zhang, and Z. Nazeri, *Community evolution in dynamic multi-mode networks*, 2008, pp. 677–685.

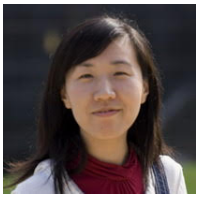
[23] C. S. Stevenson, C. Docx, and et al., "Comprehensive gene expression profiling of rat lung reveals distinct acute and chronic responses to cigarette smoke inhalation," *Am J Physiol Lung Cell Mol Physiol*, vol. 293, no. 5, pp. 1183–1193, 2007.

[24] Y. Xiang, C.-Q. Zhang, and K. Huang, "Predicting glioblastoma prognosis networks using weighted gene co-expression network analysis on tcga data." *BMC Bioinformatics*, vol. 13 Suppl 2, no. Suppl 2, pp. S12:1–8, 2012.
[25] B. Zhang and S. Horvath, "A general framework for weighted gene co-expression network analysis." *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, pp. Article17:1–37, 2005.
[26] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock, "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium." *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.
[27] N. Du, B. Wu, X. Pei, B. Wang, and L. Xu, "Community detection in large-scale social networks," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*. ACM, 2007, pp. 16–25.
[28] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," *Physical review E*, vol. 72, no. 2, p. 027104, 2005.
[29] M. E. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
[30] A. Lancichinetti and S. Fortunato, "Community detection algorithms: a comparative analysis," *Physical review E*, vol. 80, no. 5, p. 056117, 2009.
[31] P. Bogdanov, M. Mongiov X Ec, and A. K. Singh, "Mining heavy subgraphs in time-evolving networks," pp. 81–90, 2011.
[32] A. Ahmed and E. P. Xing, "Recovering time-varying networks of dependencies in social and biological studies," *Proceedings of the National Academy of Sciences*, vol. 106, no. 29, pp. 11 878–11 883, 2009.
[33] M. Takaffoli, F. Sangi, J. Fagnan, and O. R. Zaane, "Modec - modeling and detecting evolutions of communities," in *ICWSM'11*, 2011.
[34] P. Bródka, S. Saganowski, and P. Kazienko, "GED: the method for group evolution discovery in social networks," *ArXiv e-prints*, 2012.
[35] S. White and P. Smyth, "A spectral clustering approach to finding communities in graphs," *In Prof. of SDM'05*, 2005.

Dr. Nan Du received his Ph.D degree from Computer Science & Engineering department in State University of New York at Buffalo, NY, with supervision by Prof. Aidong Zhang. Prior to that, he received the BS degree from Guangdong University of Technology in 2006, and the MS degree from Southern China University of Technology in 2009. His research interests are in the areas of data mining, machine learning, and bioinformatics.



Xiaowei Jia is currently a Ph.D. candidate in computer science from State University of New York at Buffalo. He is working in data mining and machine learning with a focus on social network analysis, recommender systems and health-care signals.



Dr. Jing Gao is currently an assistant professor in the Department of Computer Science at the University at Buffalo (UB), State University of New York. She received her PhD from Computer Science Department, University of Illinois at Urbana Champaign in 2011, and subsequently joined UB in 2012. She is broadly interested in data and information analysis with a focus on information integration, crowdsourcing, ensemble methods, mining data streams, transfer learning and anomaly detection. More information about her research can be found at: http://www.cse.buffalo.edu/ jing. She is a member of IEEE.

Vishrawas Gopalakrishnan is a Ph.D. candidate in the department of Computer Science and Engineering at State University of New York at Buffalo. His research focus is on using graph techniques and machine learning in the field of text and web mining.

Dr. Aidong Zhang is SUNY Distinguished Professor and Chair in the Department of Computer Science and Engineering at State University of New York at Buffalo. Her research interests include data mining, bioinformatics, multimedia and database systems, and content-based image retrieval. She is an author of over 250 research publications in these areas. She has chaired or served on over 100 program committees of international conferences and workshops, and currently serves several journal editorial boards. She has published two books Protein Interaction Networks: Computational Analysis (Cambridge University Press, 2009) and Advanced Analysis of Gene Expression Microarray Data (World Scientific Publishing Co., Inc. 2006). Dr. Zhang is a recipient of the National Science Foundation CAREER award and State University of New York (SUNY) Chancellor's Research Recognition award. Dr. Zhang is an IEEE Fellow.