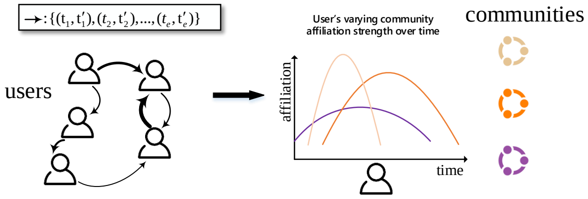
Dynamic Detection of Communities and Their Evolutions in Temporal Social Networks

Paper ID: 262

Abstract

Recently, a variety of community detection models have enjoyed impressive success. However, most of them are limited to static social networks. Snapshot-based methods for dynamic community detection also degrade when we are interested in more detailed changes in a cluster. In this paper, we propose a novel model, which explores the community evolution in temporal social networks through modeling temporal affiliation strength between users and communities. Instead of transforming dynamic networks into static networks, we propose utilizing normal distribution to estimate the changes of strength, which is more concise and comprehensive. Extensive quantitative and qualitative evaluation on large social network datasets show that our model achieves improvements in terms of prediction accuracy and reveals distinctive insight about evolutions of temporal social networks.



(a) Input

(b) Output

Figure 1: (a) is the input network where links denote people's interaction. (b) is the output bipartite affiliation graph. Each user will connect communities with a weight like F_{uA} in a normal distribution with mean value μ_{uA} and variance σ_{uA}.

Introduction

A community is a cluster of nodes with more intense interactions amongst its members than the rest of the network (Leskovec, Lang, and Mahoney 2010). Using citation network as an example of social network, millions of scholars share knowledge and make connections with each other in the means of publishing and citing papers. Such interactions between scholars exhibit their research areas of interests. We consider each scholar as a node and each research area as a community in this social network. Revealing the traits of these communities helps us understand the development of a social network, thus enabling us to answer the following questions such as:

- When does a certain user's membership to a certain community begin and fade?
- How does a community evolve over time?

Most studies in dynamic community detection, also known as *community evolution*, focus on clustering algorithms from macroscopic perspective of communities (Acar, Dunlavy, and Kolda 2015; Chakrabarti, Kumar, and Tomkins 2011; Chen et al. 2010; Tang et al. 2008), such as density (Takaffoli, Rabbany, and Zaïane 2014), and clique percolation (Palla, Barabási, and Vicsek 2007). Another line of work on dynamic community detection divides temporal networks into multiple slices (snapshots) (Yang et al. 2011; Greene, Doyle, and Cunningham 2010; Sun et al. 2010), so that a dynamic network can be treated as a time series of static networks.

Existing approaches, however, still have some drawbacks. The macroscopic perspective of a community cannot reveal the detailed evolution of communities. It can also lose much information to observe a community as a series of consecutive snapshots over time. Some approaches merge snapshots to analyze networks at a large time granularity, which can lead to inaccurate lifetime detection for a cluster because of the loss of details about the changes of clusters. In this paper, we would like to answer following research questions:

1. How can we detect the community evolutions from the perspective of individual members?
2. How can we model the consecutive membership strength between users and communities?

We answer such questions by studying community evolution through modeling temporal strength between users and communities. Figure 1 shows the overview of our model. The input is an interaction network among users, where the directed edges have two time-stamps for two users (see *Definition 1* for more details). The output is a bipartite affiliation graph where each user can connect multiple communities with scores in a distribution with respect to time.

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Modeling dynamic community strength in a continuous distribution is challenging. Simply maximizing the likelihood produces almost infinite negative edges. Our model utilizes the normal distribution to extract the community strengths, means and variances of distributions autonomously. We also propose a novel method of sampling negative edges to antagonize positive edges, which achieves a better performance.

We evaluate our model with both quantitative and qualitative evaluation. For quantitative evaluation, we apply link prediction tasks. Additionally, we quantitatively analyze the dynamic relationship between communities and users, and formation process of individual communities.

Our main contributions are as follows:

1. *Perspective*: We study community evolution by modeling temporal strength between users and communities and observe flows of the membership of users among multiple communities as community evolution.
2. *Our Model*: To the best of our knowledge, we are the first to model the continuous relationship between users and communities with normal distribution; we also propose a novel approach of sampling negative samples to maximize the likelihood. We name our model CDOT, which is short for *Community Detection Over Time*. We release all the code for the models described in this paper. [1]
3. *Performance:* Extensive experiments on large datasets show our model outperforms strong baseline methods by a large margin in terms of prediction accuracy. We also present some real-world applications of our model in academic networks.

Modeling

In this section, we first mathematically formulate our problem and then describe our modeling.

Problem Formulation

We start the formulation with introducing the background of temporal interaction network. Figure 2 illustrates the interactions among users in a social network, which can be seen as a user reposting network or a citation network . In this social network, each user interacts with others by posting a paper or tweet that mentions a post of another user. Each interaction produces a link with two time stamps as attributes: i) the first time stamp is the time when the original post is posted; ii) the second time stamp is the time when it is mentioned. By ignoring the posts, we can build a network consisting of users and there can be multiple edges with time stamps between two users. We show the definitions of some basic concepts used throughout this paper.

Definition 1. Temporal Interaction Network. A temporal interaction network is defined as a directed unweighted graph $G = (V, E)$. V is the vertex set representing $|V|$ users. The edge set E denotes the interactions between users. Each edge $e \in E$ is associated with two time stamps t_1, t_2. This can be derived from the various types of interactions such as citing papers, re-tweeting and commenting on social media. The time stamps represents the posting time (t_1) and mentioning time (t_2). A directed edge (u, t_1, v, t_2) indicates the communication of users at different moments.

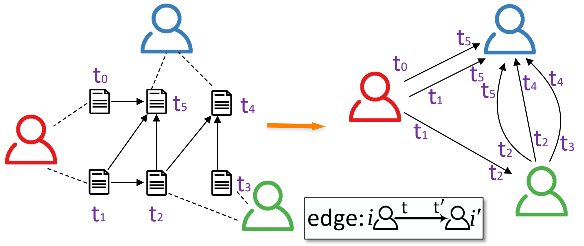


Figure 2: An illustration of social interaction.

Definition 2. Affiliation Strength. A nonnegative parameter F_{uc} represents the *affiliation strength* of a user u in a community c (Yang and Leskovec 2013). $F_{uc} = 0$ means node u is not affiliated to community c. In temporal networks, we assume that the strength between a community and a user is not fixed but changes over time. Thus, we use $P_{uc}(t)$ to represent the weights of F_{uc} at time t. Finally, we have

$$\pi_{uc}(t) = P_{uc}(t)F_{uc}$$

as the temporal strength between a user u in a community c at time t.

Definition 3. Probability of User Interaction. We denote p_{u,v,t_1,t_2} as the probability of the existence of an edge (u, t_1, v, t_2) in a *Temporal Interaction Network*. We assume the connection between users are through all internal communities with different contributions. The probability of an interaction between two users through a particular community c is $\pi_{uc}(t_1)\pi_{vc}(t_2)$. $p(u, t_1, v, t_2)$ is calculated in the following equations, where $\mathcal{C}$ is the set of all communities.

$$\begin{aligned} H &= \sum_{c \in \mathcal{C}} \pi_{uc}(t_1)\pi_{vc}(t_2), \\ p(u, v, t_1, t_2) &= 1 - \exp(-H), \end{aligned} \tag{1}$$

The problem we aim to solve is how to better model the such probability of edges in temporal social networks.

Model Description

Figure 1 in *Introduction* shows the structure of our model. Given a temporal social network, our model produces a bipartite graph where the nodes on one side represent the *users* in the social network G and the nodes on the other side represent *communities*. Denote the time t as *optimal time* when the $\pi_{uc}(t)$ achieves its maximum value. We assume the distribution of the optimal time t is *normal distribution*. Thus, we have μ_{uc} and σ_{uc} as the mean and the variance values of the normal distribution with respect to $\pi_{uc}(t)$. In the following paragraphs, we talk about the reasons why we choose normal distribution and the effects.

1. We argue that it is very common that a user u enters into a field of interest (a community c) at a particular time. Finally, at another time, it leaves from this community forever. Thus, $\pi_{uc}(t)$ tends to be unimodal.

---

[1] `http://anoymous.due.to.blind.review/`

2. The two parameters are very reasonable. The more a time stamp t is close to the mean μ, the higher affiliation strength $\pi_{uc}(t)$ will be. σ_{uc} represents the duration of the membership of u with c to some extent, A small σ_{uc} suggests the affiliation is more *dynamic* and just appears at a particular short period of time near μ_{uc}. A larger σ_{uc} indicates the affiliation is more *stable* and lasts for a longer time.
3. Another reason is that, using normal distribution just incorporates only two new parameters for each affiliation and can model the strength at any time. Whereas, snapshot-based method demands us to control the window numbers, which is very hard in practice.
4. Moreover, it is easier and more efficient to compute the derivatives when we use normal distribution than using other distributions. While tuning the parameters of the normal distribution, we can calculate only one parameter at a time. The gradient function is then convex, which make the optimization feasible. Related formulas and deductions are discussed in the *Parameter Learning* section.

With this assumption, our model can capture more information from temporal individual relationship and represent the changes of affiliation strength more easily. It is true that there are some successful attempts in community detection, including Cluster Affiliation Model for Big Networks (BIGCLAM) (Yang and Leskovec 2013) over networks, and COmmunity Level Diffusion (COLD) (Hu et al. 2015) over text, time and networks. Our model significantly goes beyond those by directly studying the temporal components with a distribution perspective. The parameter training process is discussed in the next section.

Parameter Learning

In this section, we explain the parameters learning process of our model.

Objective Function

Given the interaction network $G(V, E)$, we aim to detect K communities by fitting our model to the underlying network G. Our goal is to maximize the log-likelihood $l(F, \mu, \sigma) = \log P(G|F, \mu, \sigma)$:

$$\hat{F}, \hat{\mu}, \hat{\sigma} = \underset{F \geq 0, \sigma > 0}{\arg\max}\, l(F, \mu, \sigma), \tag{2}$$

where H is defined in Eq. 1 and

$$l(F, \mu, \sigma) = \sum_{(u,v,t_1,t_2) \in E} log(1 - \exp(-H)) - \sum_{(u,v,t_1,t_2) \notin E} H. \tag{3}$$

Sampling

As a matter of fact, we notice that the second term in the likelihood is almost infinite because the time t is a continuous parameter; even if we divide the time interval into slices, the number of parameters will be enormous. Therefore, we adopt the sampling method to restrict the amount of negative edges.

$N^+(u)$ is a set of edges connecting to node u, and $N^-(u)$ is a set of negative edges that we randomly sample with uniform distribution from the nonexistent edges in origin network G. As for the t_1 and t_2, we sample them by first setting two time ranges respectively for u and v when their interactions occur the most. The number of negative edges equals to a definite ratio r of the original edges. Finally, we can compute $\sum_{(v,t_2) \in N^-(u)} F_{vc}$ with time complexity $O(N^+(u) \times r)$. Thanks to the extreme sparsity of real-world networks, we are able to update F_u, μ_u, σ_u in near constant time.

Computing Derivatives

To optimize our objective function defined in Eq. 2, we adopt the block coordinate gradient ascent method (Hsieh and Dhillon 2011). Before any computation, we initialize our F using the result of BIGCLAM(Yang and Leskovec 2013) with our sampling method, so that the optimization is faster.

We denote $\mathbf{F_u}$ as a vector consisting of $\{F_{uc}|c \in \mathcal{C}\}$, representing the affiliation strength of the user u to all the communities. We update $\mathbf{F_u}$ for each node with all other parameters being fixed. The mean value $\boldsymbol{\mu}_{\text{u}}$ and the variance σ_u are updated in the similar way. We utilize this updating method since if we fix the other parameters, the problem of optimization will convert to a convex optimization problem. Then we solve three following subproblems.

Subproblem 1*: updating $F_u c$ with fixed μ_{vc} and σ_{vc}*

$$\arg\max\ l(\mathbf{F_u}), \tag{4}$$

where

$$l(\mathbf{F_u}) = \sum_{(u,v,t_1,t_2) \in N^+(u)} \log(1 - \exp(-H)) - \sum_{(u,v,t_1,t_2) \in N^-(u)} H. \tag{5}$$

The subproblem can be further solved by projected gradient ascent,

$$F_{uc}^{new} = \max(0, F_{uc}^{old} + \alpha_{F_u} \nabla(F_{uc})), \tag{6}$$

where α_{F_u} is the step size computed by backtracking line search, and the gradient is:

$$\begin{aligned} \nabla l(F_{uc}) = &\sum_{(v,t_2) \in N^+(u)} \frac{\exp(-H)}{1 - \exp(-H)} P_{uc}(t_1) \pi_{vc}(t_2) \\ &- \sum_{(v,t_2) \in N^-(u)} P_{uc}(t_1) \pi_{vc}(t_2). \end{aligned} \tag{7}$$

Subproblem 2*: updating $\mu_u c$ with fixed F_{vc} and σ_{vc}*

After the community affiliation matrix $\boldsymbol{F}$ updated, we fix $\boldsymbol{F}$ and $\boldsymbol{\sigma}$,update the community time parameter matrix $\boldsymbol{\mu}$:

$$\mu_{uc}^{new} = \max(0, \mu_{uc}^{old} + \alpha_{\mu_u} \nabla(\mu_{uc})). \tag{8}$$

where α_{μ_u} is computed as α_{F_u}, and the gradient is:

$$\begin{aligned} \nabla l(\mu_{uc}) = &\sum_{(v,t_2) \in N^+(u)} \frac{\exp(-H)}{1 - \exp(-H)} \pi_{uc}(t_1) \pi_{vc}(t_2) \frac{t_1 - \mu_{uc}}{\sigma_{uc}^2} \\ &- \sum_{(v,t_2) \in N^-(u)} \pi_{uc}(t_1) \pi_{vc}(t_2) \frac{t_1 - \mu_{uc}}{\sigma_{uc}^2}. \end{aligned} \tag{9}$$

Subproblem 3*: updating $\mu_u c$ with fixed F_{vc} and σ_{vc}*

After updating the community time parameter matrix $\boldsymbol{\mu}$, we fix $\boldsymbol{F}$ and $\boldsymbol{\mu}$, update the community time parameter matrix $\boldsymbol{\sigma}$:

$$\sigma_{uc}^{new} = \max\left(0, \sigma_{uc}^{old} + \alpha_{\sigma_u} \nabla(\sigma_{uc})\right), \tag{10}$$

where

$$\begin{aligned} \nabla l(\sigma_{uc}) = & \sum_{(v,t_2)\in N^+(u)} \frac{\exp(-H)}{1-\exp(-H)} \pi_{uc}(t_1)\pi_{vc}(t_2) \frac{(t_1-\mu_{uc})^2 - \sigma_{uc}^2}{\sigma_{uc}^3} \\ & - \sum_{(v,t_2)\in N^-(u)} \pi_{uc}(t_1)\pi_{vc}(t_2) \frac{(t_1-\mu_{uc})^2 - \sigma_{uc}^2}{\sigma_{uc}^3}. \end{aligned} \tag{11}$$

The whole process is illustrated in Algorithm 1.

Algorithm 1 Parameter learning for our model

```
Initialize $\boldsymbol{F}, \boldsymbol{\mu}, \boldsymbol{\sigma}$.
repeat
    Sample negative edges in a definite ratio of origin edges
    for $u = 1, 2, ..., N$ do
        Calculate $\nabla l(F_u)$ based on Eq.7
        Calculate $\alpha F_u$ using backtracking line search
        Update $F_u$
    for $u = 1, 2, ..., N$ do
        Calculate $\nabla l(\mu_u)$ based on Eq.9
        Calculate $\alpha \mu_u$ using backtracking line search
        Update $\mu_u$
    for $u = 1, 2, ..., N$ do
        Calculate $\nabla l(\sigma_u)$ based on Eq.11
        Calculate $\alpha \sigma_u$ using backtracking line search
        Update $\sigma_u$
until convergence or $max_{iter}$ is reached
Return parameters $F, \mu, \sigma$.
```

Time Complexity Our model involves three times the number of parameters than BIGCLAM (Yang and Leskovec 2013), as we consider the temporal effects. Whereas, we achieve a time complexity at the same level of BIGCLAM. Our model has higher speed than COLD (Hu et al. 2015), which is based on Gibbs sampling and thus needs to run in a particular order. A notable advantage of our model is that it can compute each node *simultaneously* since it uses block coordinate gradient ascent method.

Evaluation

We evaluate our model from various aspects. We first quantitatively evaluate the model with extracting overlapping communities in a link prediction task. Similarly, we test the performance of time prediction. Finally, we qualitatively evaluate our model with showing temporal relation between communities and users by several cases.

Evaluation Setup

Datasets Our datasets are extracted from MAG (Microsoft Academic Graph) (Sinha et al. 2015), a citation network including over 160M papers. Due to the enormous amount of nodes and edges, we selected part of papers and their corresponding authors. We create two datasets (M200, BD) for our quantitative evaluation and qualitative evaluation respectively.

The quantitative evaluation dataset consists of papers from 200 academic conferences under computer science, named as *M200*. Based on these papers, we gain their authors and years. For authors, we have their fields of study which serve as the ground truth of overlapping communities. Additionally, we extract paper titles, which is necessary for COmmunity Level Diffusion(COLD). The subgraph of MAG is built by these features where nodes are authors and directed edges are citation relationships with a pair of publishing years as an attribute, (u, t_1, v, t_2) (recall *Definition 1*). We divide it into two parts: 80% is training set and 20% is the testing set. We utilize 5-fold cross validation on this dataset in training process. Note that this division is only applied on edges where nodes are unchanged. We randomly select all edges between two nodes which still have edges to other nodes if selected edges were removed, in order to keep the local connectivity that each node is connected to a part of the graph. Mathematically, the method keeps the number of connected components unchanged before and after division.

Another dataset is used for qualitative evaluation consists of papers under the research topic *big data*, named as BD. Features of authors and titles and the way of building subgraph are the same as above. We apply our model to the whole set without division, aiming to give a overall picture of the affiliation of each author in his/her involved research areas under BD.

The statistics of the dataset is shown in Table1.

Dataset	Nodes	Edges	Papers	Ground-truth Communities
M200	318915	4093867	500869	19 and 290
BD	81488	2505560	120268	25

Table 1: Statics of the M200 and BD dataset.

Baseline Methods We compare the proposed model with several following state-of-the-art competitors.

1. Cluster Affiliation Model for Big Networks **(BIGCLAM)** (Yang and Leskovec 2013) only uses edges in networks based on affiliation graph model. In *Parameter Learning* section, we improve BIGCLAM with sampling method, otherwise, its performance is far from being compared. This modified model is marked as BIGCLAM*.
2. Communities through Directed affiliations **(CoDA)** (Yang, McAuley, and Leskovec 2014) is also improved from BIGCLAM. It divides a community into two patterns: cohesive or 2-mode. The feature accounts for edge directions.
3. COmmunity Level Diffusion **(COLD)** (Hu et al. 2015) models topics and communities in a unified latent framework. It addresses the community diffusion problem over text, time and networks. Except for link prediction, it uses topics and memberships to accomplish the time stamp prediction task.

Note that our model, CoDA and BIGCLAM take only nodes and edges as input, while COLD additionally utilizes titles of papers. There are differences in applications between these four models. All of four have the ability to detect communities and make a prediction of the existence of an edge. Moreover, our model and COLD are able to predict a particular time of a establishment. Utilizations of features and applications of models are summarized in Table 2. All experiments are conducted on a Linux machine with 40 2.4GHz CPU cores and 128G memory.

	Features			Tasks	
Model	Graph	Time	Titles	Community Detection	Time Prediction
CDOT	○	○	×	○	○
CoDA	○	×	×	○	×
BIGCLAM*	○	×	×	○	×
COLD	○	○	○	○	○

Table 2: Features and Tasks of Models

Model Initialization We find that BIGCLAM(Yang and Leskovec 2013) [2] has a serious problem. If K is small, each community is relatively big in a huge network. Consequently, all the affiliation strength between users and communities is near zero. The main reason is that BIGCLAM assumes that two nodes affiliate to a same community only if they have edges. However, this constraint is too strict, because few people could connect all the nodes in a large community except for some people who are (or close to) the center of this community. That is why BIGCLAM can only find small groups where people connect with each other more densely. When we apply this method to large social networks, where anyone may cite anybody and the fields are bigger, containing more users, the original BIGCLAM dose not work. We improve BIGCLAM with the sampling method. Then use the results of it to initialize the strength of our model F. It speeds up the inference process and saves us a lot of time. After that, for each node u and community c whose $F_{uc}>0$, we initialize μ_{uc} (the midpoint of the whole period T) and $\sigma_{uc} = \frac{1}{2}T$.

Sampling Ratio In our experiment we find that the sampling ratio is very important. If the number of communities we want to detect is small, we set the ratio larger, in order to avoid the situation that one user lies in excess number of communities. When the number of communities is large, we set the ratio small. Then the communities spread from initial seeds. In our experiment below, we set the sampling ratio 0.5 for 19 communities, and 1.0 for 290 communities.

Choose the number of communities. We set the number of communities as ground truth. Although there may be multiple divisions of communities in a network, we can compare with the other baselines better in this condition.

Quantitative Evaluation – Prediction

The quantitative evaluation part is based on the dataset M200. We first evaluate our model based on *link prediction*. Then, we test temporal modeling by *time stamp prediction*.

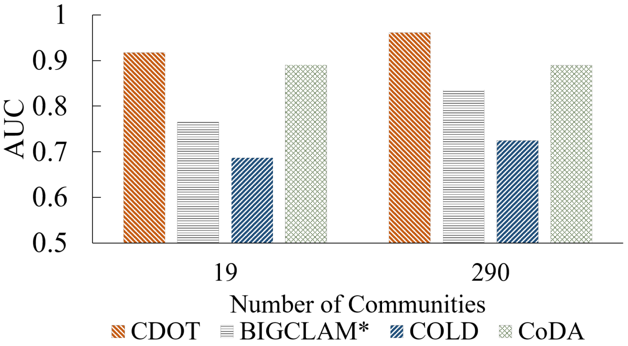


Figure 3: Link Prediction Performance

Community Detection (Link Prediction) A common method measuring the performance of a model in community detection task is link prediction (Liben-Nowell and Kleinberg 2007). Given a set of edges, the model estimates the probability that they are links between users. In our model, the probability from user u at time t_1 to user v at time t_2 is measured by Eq. 1.

Area under the receiver operating characteristic curve (AUC) is taken as the prediction measurement, since designating a threshold for the probability for link existence is artificial. An edge is called positive when it comes from the original dataset. Here we take all edges in the testing set as positive edges. Accordingly, negative edges are ones which do not exist in the original dataset. We randomly select negative edges equal to 10% of the number of edges in training set. All of models take the number of communities as an input parameter. It is assigned as the exact number of ground truth.

Figure 3 shows the AUC scores of four models. Our model demonstrates better performance on link prediction task than other models. Moreover, BIGCLAM* and CoDA is significantly more accurate than COLD, showing that affiliation graphic model outperforms topic models in academic society. We infer the reason is that, unlike Twitter or Weibo, the structure of academic society has more restrictions. People in academic society can not say anything at any time. The better performance of our model than BIGCLAM* and CoDA reveals that it is feasible for our model to capture dynamic strength between users and communities.

Temporal Modeling (Time Stamp Prediction) Here we demonstrate the novel feature of our model that shows superiority over existing methods in terms of time prediction. Time stamp prediction (Wang and McCallum 2006) is to estimate the occurring time of a previously unseen document. Given a user u, another user v who cites u, and the publish time of original document t_1, which are three of four elements in a link (u, t_1, v, t_2), the model predicts the value of t_2. It measures the capability of modeling the distribution of activation time of an author. Here our method to measure time prediction is negative log-likelihood (N-log). We

---

[2] SNAP:`http://snap.stanford.edu/snap`

calculate all of the likelihoods

$$p(u, t_1, v, t_2^{'}) = \sum_{c=0}^{K} F_{uc} P_{uc}(t_1) F_{vc} P_{vc}(t_2^{'})$$

for all time stamps $t_2^{'} \in [mintime, maxtime]$, where $mintime$, $maxtime$ are the minimum and maximum of publish time in the dataset respectively. We then apply normalization

$$p_n(u, t_1, v, t_2^{'}) = \frac{p(u, t_1, v, t_2^{'})}{\sum_{t_2^{'}=mintime}^{t_2^{'}=maxtime} p(u, t_1, v, t_2^{'})}$$

to all $p(u, t_1, v, t_2^{'})$. Next, we calculate negative log-likelihood $l = -\log p_n(u, t_1, v, t_2)$ for truth value t_2. We take the average value of all negative log-likelihoods from input edges as the result of nlog measurement. Mention that, we set a upper threshold for the value of negative log-likelihood. We reset all the values greater than 5 to 5, in order to measure which model is capable to predict more accurately. Thus, the result of nlog measurement is a positive value less than 5. Note that since BIGCLAM(*) and CoDA do not support the temporal prediction.

Model	19 Communities	290 Communities
CDOT	**2.734**	2.363
COLD	4.693	**2.161**

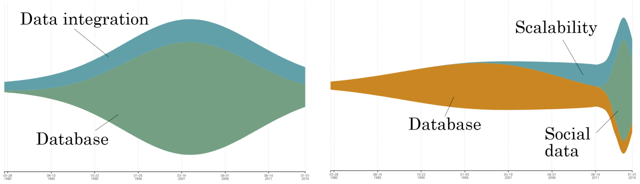
Table 3: Nlog Measurement Result

Table 3 illustrates the score of nlog measurement. Our model dramatically outperforms COLD in predicting time stamp of community 19, but COLD is slightly better than our model in community 290. One significant point is that our model does not utilize title information compared to COLD. Our model has the superiority in simpleness without losing prediction accuracy. It directly makes the assumption that strength of a user obeys normal distribution over time, which gains considerable veracity on predicting publish year of an unseen paper. In contrast, COLD adopts multinomial distributions over topics and time, where its complexity on modeling is higher. We argue that in this example, COLD runs 45.7k second while ours only needs 2.4k seconds. Also, they utilize more textual information while we only consider the network.

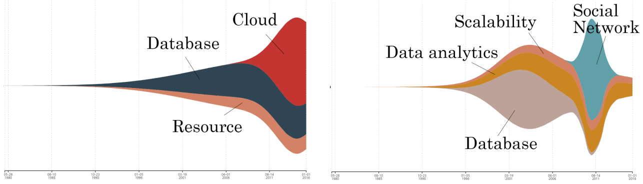
Qualitative Evaluation – Application

Utilizing the temporal social network, our model is able to detect users' strength of affiliation among truth or latent communities. Furthermore, we can demonstrate users' belonging to communities at a time to grasp the whole picture of communities. In this part, we associate title and author information with our result for visualization analytics, therefore we can demonstrate related names and keywords, which is not used in model training. This part of experiment is base on the dataset BD.

Since the likelihood of the link between a user and a community represents the affiliation strength, we regard the likelihood as the activation of a researcher in such field of study.



(a) Joann J. Ordille (b) Stefano Ceri

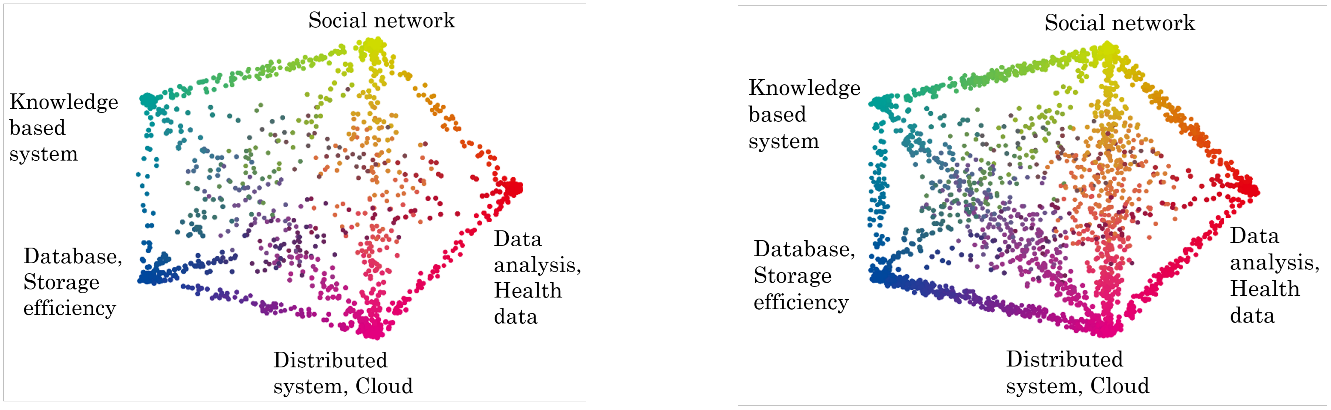


(c) Danilo Ardagna (d) Hakan Ferhatosmanoglu

Figure 4: Temporal activation of researcheres among fields. Names are listed below the corresponding pictures. The horizontal axis represents the time, from 1980 to 2016. The vertical one suggestes the affiliation strength. Each block(color) is on behalf of a community, or a field of study equally. The overall wavy shape reflects the variation of affliation strength through one's research career.

We map likelihoods of all communities in the same year to $[0, 1]$ according to their propotions of the sum of them. Figure 4 consists visualizations of temporal activation variation by time among fields under topic *big data*, from 4 researchers. Figure 4(a) gives the most common and simplest snapshot of a research career, concentrated on one or two research topics, with stages of entry level, high production and exit. Figures 4(b) to 4(d) demonstrate the situation that a researcher explors new research field in the middle of his career. There are vanishment of activation in communities in Figure 4(d). From these visualizations of temporal activation, including appearance of new fields, vanishment of past fields and steady state of ongoing research, we have certitude that our model has its distinctive capability of temporal modeling. The superiority of our model lies in the ability to predict the distribution with less features even missing data.

Here we illustrate the feature from the large scale perspective of communities. Figure 5(a) and 5(b) are visualizations of selected communities in 1987 and 2016 respectively. The visualization result either demonstrates the capability of community detection. Moreover, we observe the significant phenomenon about community evolution. Compared with the picture in 1987, we can find that there are few times of academic interactions between research fields more than before. An acceptable explanation is that the topic *big data* was a prospective study in 1987 but it has become one of most leading edge studies since the twenty-first century, especially the second decade. The actual demands of industry propose large amount of practical problems, which are frequently complex, for scientists to collaborate with each other coming up with solutions. This actuates the ongoing development of a field and incremental researchers getting involved in multiple areas and crossing fields. A common feature of two pictures is that points lying between corners



(a) 1987

(b) 2016

Figure 5: Communities in research field *big data*. We select 2 most active communities of a researcher in a specific year. A researcher is more active in a community while the colored point is getting more closed to the corner and has a bigger size. Points align on sides and diagonal lines represent researchers active in both two communities with slight disparity. We choose 5 communities with most number of researchers and omit others, intending to draw a distinct picture between communities and years. Related keywords are listed aside each of the corners.

are distributed approximatively uniformly, indicating that 5 communities here have their specialty without becoming an appendage of others, although interactions are abundant. Thus, this picture shows a piece of community evolution concept due to our continous temporal modeling.

Related Work

The community detection work can be divided into two types. The first type is *static community detection*, and the second one is *dynamic community detection*, also known as *community evolution*.

Static Community Detection

Static community detection has been extensively investigated in the last decade (Xie, Kelley, and Szymanski 2013). A series of affiliation graph models (Yang and Leskovec 2012; 2013) are proposed based on the idea that communities arise due to shared group affiliations (Breiger 1974). Meanwhile, a increasing number of recent works incorporate both the network structure and content to improve community detection performance, such as Link-PLSA-LDA (Nallapati et al. 2008), PMTLM (Zhu et al. 2013), and COLD (Hu et al. 2015). This line of work, however, defines the communities are static. This can be limited to volatile users. Instead, we propose to model the relationship between users and communities in a dynamic way, which better fits the real-world application and reveal more information about the evolution of temporal social networks.

Dynamic Community Detection

Existing work of dynamic community detection can also be divided into two main kinds of approaches according to the information they leverage: *Indirect methods* and *Direct methods*.

Indirect methods firstly focus on identifying communities within a set of *snapshots* of target networks, and then synthesize a final model of their lifetime based on the time steps(Acar, Dunlavy, and Kolda 2015; Chakrabarti, Kumar, and Tomkins 2011; Chen et al. 2010; Tang et al. 2008). In each snapshot of the network, temporal factors are thought as the same and thus they can utilize conventional static community detection methods. Their results of lifetime of communities can be inaccurate when when network snapshots are sparse and contain few connections. Some methods merge snapshots to analyze data at a large time granularity lose the details of the change in a cluster.

Direct methods design models with temporal factors to directly leverage the whole information in time series. Consequently, they can learn the community structure from the original dynamic network(Fu, Song, and Xing 2009; Leonardi et al. 2016; Mao et al. 2014). Our method is one of the direct methods as well. Most of the direct methods are based on tensor decomposition techniques, resulting in low physical interpretation. By comparison, our work identifies communities in a continuous distribution, which provides a more detailed and comprehensive view.

Conclusion

We propose the model CDOT (Community Detection Model of Multi-interaction Over Time) to first represent the temporal factors regarding the strength between users and communities in temporal social networks and then to detect dynamic communities. CDOT uses Normal distribution so it can avoid some deficiencies that could be brought by using static snapshots. Both experiments and applications validate that our model achieves substantial improvements to the baseline methods.

References

Acar, E.; Dunlavy, D. M.; and Kolda, T. G. 2015. A scalable optimization approach for fitting canonical tensor decompositions. *Journal of Chemometrics* 25(2):67–86.

Breiger, R. L. 1974. The duality of persons and groups. *Social forces* 53(2):181–190.

Chakrabarti, D.; Kumar, R.; and Tomkins, A. 2011. Evolutionary clustering. In *Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, Pa, Usa, August*, 554–560.

Chen, Y. C.; Rosensweig, E.; Kurose, J.; and Towsley, D. 2010. Group detection in mobility traces. In *International Wireless Communications and Mobile Computing Conference, Iwcmc 2010, Caen, France, June 28 - July*, 875–879.

Fu, W.; Song, L.; and Xing, E. P. 2009. Dynamic mixed membership blockmodel for evolving networks. In *International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June*, 329–336.

Greene, D.; Doyle, D.; and Cunningham, P. 2010. Tracking the evolution of communities in dynamic social networks. In *Advances in social networks analysis and mining (ASONAM), 2010 international conference on*, 176–183. IEEE.

Hsieh, C.-J., and Dhillon, I. S. 2011. Fast coordinate descent methods with variable selection for non-negative matrix factorization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1064–1072. ACM.

Hu, Z.; Yao, J.; Cui, B.; and Xing, E. 2015. Community level diffusion extraction. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 1555–1569. ACM.

Leonardi, S.; Anagnostopoulos, A.; cki, J.; Lattanzi, S.; and Mahdian, M. 2016. Community detection on evolving graphs.

Leskovec, J.; Lang, K. J.; and Mahoney, M. 2010. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th international conference on World wide web*, 631–640. ACM.

Liben-Nowell, D., and Kleinberg, J. 2007. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology* 58(7):1019–1031.

Mao, H. H.; Wu, C. J.; Papalexakis, E. E.; Faloutsos, C.; Lee, K. C.; and Kao, T. C. 2014. *MalSpot: Multi 2 Malicious Network Behavior Patterns Analysis*. Springer International Publishing.

Nallapati, R. M.; Ahmed, A.; Xing, E. P.; and Cohen, W. W. 2008. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 542–550. ACM.

Palla, G.; Barabási, A.-L.; and Vicsek, T. 2007. Quantifying social group evolution. *arXiv preprint arXiv:0704.0744*.

Sinha, A.; Shen, Z.; Song, Y.; Ma, H.; Eide, D.; Hsu, B.-j. P.; and Wang, K. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, 243–246. ACM.

Sun, Y.; Tang, J.; Han, J.; Gupta, M.; and Zhao, B. 2010. Community evolution detection in dynamic heterogeneous information networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, 137–146. ACM.

Takaffoli, M.; Rabbany, R.; and Zaïane, O. R. 2014. Community evolution prediction in dynamic social networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*, 9–16. IEEE.

Tang, L.; Liu, H.; Zhang, J.; and Nazeri, Z. 2008. Community evolution in dynamic multi-mode networks. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 677–685.

Wang, X., and McCallum, A. 2006. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 424–433. ACM.

Xie, J.; Kelley, S.; and Szymanski, B. K. 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)* 45(4):43.

Yang, J., and Leskovec, J. 2012. Community-affiliation graph model for overlapping network community detection. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, 1170–1175. IEEE.

Yang, J., and Leskovec, J. 2013. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 587–596. ACM.

Yang, T.; Chi, Y.; Zhu, S.; Gong, Y.; and Jin, R. 2011. Detecting communities and their evolutions in dynamic social network bayesian approach. *Machine Learning* 82(2):157–189.

Yang, J.; McAuley, J.; and Leskovec, J. 2014. Detecting cohesive and 2-mode communities indirected and undirected networks. In *Proceedings of the 7th ACM international conference on Web search and data mining*, 323–332. ACM.

Zhu, Y.; Yan, X.; Getoor, L.; and Moore, C. 2013. Scalable text and link analysis with mixed-topic link models. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 473–481. ACM.