2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable
Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)

Boosting Temporal Community Detection via Modeling Community Evolution Characteristics

1st Wei Yu
College of Intelligence and Computing
Tianjin University
Tianjin, China
weiyu@tju.edu.cn

2nd Wenjun Wang
College of Intelligence and Computing
Tianjin University
Tianjin, China
wjwang@tju.edu.cn

3rd Xue Chen
Law School
Tianjin University
Tianjin, China
Corresponding author: xuechen@tju.edu.cn

4th Huaming Wu
Center for Applied Mathematics
Tianjin University
Tianjin, China
whming@tju.edu.cn

5th Minghu Tang
College of Intelligence and Computing
Tianjin University
Tianjin, China
School of Computer Science and Technology
Qinghai Nationalities University
Qinghai, China
mhtang@tju.edu.cn

6th Yang Yu
College of Intelligence and Computing
Tianjin University
Tianjin, China
yuyangyy@tju.edu.cn

***Abstract*—Community structure analysis in dynamic network is widely concerned in various fields, which mainly focus on temporal community detection and community evolution analysis. Most of the related works usually fist detect communities and then analyze evolution. This leads to a loss of evolution information on temporal community detection because block structures and evolution characteristics coexist in dynamic networks. Even thought a few model-based approaches consider the evolution characteristics into community detection, they need to know the number of communities in advance and ignore automatic determination of the number of communities, which is a model selection problem. In the paper, we propose an model, Evolutionary Bayesian Non-negative Matrix Factorization (EvoBNMF), to model community structures with evolution characteristics for boosting the performance of temporal community detection. In detail, EvoBNMF introduces evolution behaviors, which quantify the transition relationships of communities between adjacent snapshots, to describe the evolution characteristics of community structure. Innovatively, EvoBNMF can catch the most appropriate number of communities autonomously by shrinking the corresponding evolution behaviors. Experimental results from synthetic networks and real-world networks over several state-of-the-art methods show that our approach has superior performance on temporal community detection with the virtue of autonomous determination of the number of communities.**

***Keywords*-Bayesian Non-negative Matrix; community detection; evolution analysis; dynamic networks**

I. Introduction

Dynamic network analysis draws great attention recently for studying complex systems in lots of fields, such as physics science , biological science, information science and so on [1]. A most important research problem of dynamic network analysis is community structure analysis including temporal community detection [1] and community evolution analysis [2]. Temporal community detection helps to mine the meaningful groups or functional modules hiding in networks. Community evolution analysis exposes the evolution behaviors which quantify the transition relationships of communities between adjacent snapshots, and helps to tracing the change trends of dynamic networks. Most of the related works usually fist detect communities and then analyze evolution by matching the communities between adjacent snapshots, which lead to a loss of evolution information on temporal community detection because block structures and evolution characteristics coexist in dynamic networks. Therefore, it is very necessary to model the community structures with evolution characteristics for boosting the performance of temporal community detection.

Temporal community detection can be mainly summarized as two types: two-stage approaches [3] and evolutionary clustering based approaches [4]. Firstly, two-stage approaches first detect clusters at each snapshot using a static method and then match them across different snapshots. These approaches detect the communities of the current snapshot ignoring the historical community structures from last snapshots, which take away the evolution characteristics of temporal community structures and are usually sensitive to noise. Evolutionary clustering based approaches effectively make up for this shortcoming, which detect communities at the current snapshot using not only the current topology but also the previous community structures. However, most of these approaches ignore determining the number of communities at each snapshot automatically and need to

978-1-7281-4328-6/19/$31.00 ©2019 IEEE
DOI 10.1109/ISPA-BDCloud-SustainCom-SocialCom48970.2019.00185

be specified in advance. This is a model selection problem which is a common challenge for community detection. In addition, most of these approaches just focus on identifying temporal communities accurately but ignore analyzing the corresponding evolution.

Community evolution analysis, which is usually used to tracing the evolution behavior of community structures, is as important as temporal community detection. The existing research is mainly divided into heuristic based [3], feature engineering based approaches [5] and generative model based approaches [6]. Heuristic based approaches usually summarize the changing laws over time for analyzing the evolution after detecting communities. Feature engineering based approaches extract the evolution feature based on the detected temporal communities with machine learning algorithm. The first two types of methods tend to analyze the evolution laws after detecting communities, so that the results of the evolution analysis rely too much on the results of community detection. They ignore that community structures and evolution characteristics coexist in dynamic networks. Fortunately, generative model based approaches, which model the generative mechanism of community structure and community evolution synchronously, is able to make up for that shortcoming. However, most of these approaches just describe the evolution behaviors qualitatively but not quantitatively. And few of these approaches can deal with model selection problem automatically.

For addressing the above issues, we focus on how to model community structures with evolution characteristics for boosting temporal community detection and determining autonomously the number of communities at each snapshot of dynamic networks synchronously. In this paper, we propose an Evolutionary Non-negative Matrix Factorization (EvoBNMF) model based on a Bayesian probabilistic model. In detail, we introduce the evolution behaviors to model the evolution characteristics of community structures with Bayesian non-negative matrix factorization (BNMF) [7] in an evolutionary clustering framework [4]. Then we derive a gradient descent algorithm to maximize the posterior estimate for optimizing the parameters of our proposed model. Innovatively, our proposed EvoBNMF catches the most appropriate number of communities autonomously by shrinking the corresponding evolution behaviors of each snapshot network. Last but not least, experimental results from synthetic networks and real-world networks over several state-of-the-art methods show that our approach has superior performance on temporal community detection with the virtue of autonomous determination of the number of communities. It is worthwhile to highlight several contributions of this work here:

- We propose an Evolutionary Non-negative Matrix Factorization (EvoBNMF) model by modeling commubity detection with evolution characteristics for improving temporal community detection.
- The proposed EvoBNMF catches the most appropriate number of communities autonomously by shrinking the corresponding evolution behaviors of each snapshot network.
- An effective algorithm is developed to optimize the objective function of EvoBNMF, of which the time complexity can be degraded to be linear.
- Extensive experiments from synthetic and real-world dynamic networks demonstrate that our method has superior performance on temporal community detection in comparison with state-of-art methods.

II. Methodology

A. Notations

We set a dynamic network as a series of network snapshots. Let $\mathcal{G} = (V^{(t)}, E^{(t)}), t \in [1, 2, ..., T]$ be a dynamic undirected network, where $V^{(t)}) = \{1, ..., N^{(t)}\}$ is entity or node sets, $E^{(t)}$ is edge sets at snapshot t and T is the number of snapshots. Network snapshot t is represented with a $N^{(t)} \times N^{(t)}$ adjacency matrix $A^{(t)}$, where the element at snapshot t

$$A_{ij}^{(t)} = \begin{cases} 1 & (i,j) \in E^{(t)} \\ 0 & (i,j) \notin E^{(t)}. \end{cases}$$

In addition, we summarize the main notations in table I.

Table I
Table of Notations

Symbol	Definition
t	the snapshot label, and t $\in [1, T]$;
$V^{(t)}$	the node sets of snapshot t, and $V^{(t)} = \{1, 2, \cdots, N^{(t)}\}$;
$E^{(t)}$	the edge sets of snapshot t;
$A^{(t)}$	the adjacent matrix at snapshot t, and $A^{(t)} \in R_{+}^{N^{(t)} \times N^{(t)}}$;
$W^{(t)}$	the basis matrix at t, and $W^{(t)} \in R_{+}^{N^{(t)} \times K^{(t)}}$;
$H^{(t)}$	the community membership matrix at t, and $H^{(t)} \in R_{+}^{K^{(t)} \times N^{(t)}}$;
$Z^{(t)}$	the evolution behavior matrix at snapshot t, and $Z^{(t)} \in R^{K^{(t)} \times K^{(t-1)}}$;
$C^{(t)}$	the community label matrix of snapshot t;
$K^{(t)}$	the number of communities of snapshot t.

B. EvoBNMF Model

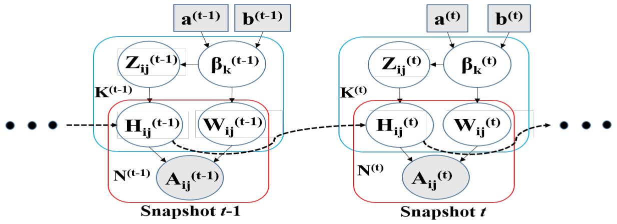

Figure 1. Graphical model of the proposed EvoBNMF.

Inspired by BNMF [7] in static networks, we use a Bayesian probabilistic model to design the generative graphical model of EvoBNMF considering the core idea of evolutionary clustering (see as Fig. 1). For snapshot 1, the graphical model can be constructed similarly according to the static BNMF because there is no historical structure information. Similar to Ref. [7], we assume that $\hat{A}_{ij}^{(1)}$ is drawn from a Poisson distribution with rate $\hat{A}_{ij}^{(1)} = \sum_{k=1}^{K^{(1)}} W_{ik}^{(1)} H_{kj}^{(1)}$. And $W_{ik}^{(1)}$ and $H_{kj}^{(1)}$ are both drawn from a half normal distribution with scale parameters $\boldsymbol{\beta}^{(1)} = \{\beta_k^{(1)}\}$. In addition, we consider the conjugate prior of half normal distribution is Gamma distribution. Therefore $\boldsymbol{\beta}^{(1)}$ is drawn from Gamma distribution with two hyper-parameters $a^{(1)}$ and $b^{(1)}$. According to the graphical model in Fig.1, the model of snapshot 1 is same as Ref. [7], and the corresponding objective function of snapshot 1 is as follows:

$$\begin{aligned}L^{(1)} = &-\sum_{i=1}^{N^{(1)}}\sum_{j=1}^{N^{(1)}}(A_{ij}^{(1)}\log\frac{A_{ij}^{(1)}}{\hat{A}_{ij}^{(1)}} + \hat{A}_{ij}^{(1)} - A_{ij}^{(1)})\\ &+\frac{1}{2}\sum_{k=1}^{K}(\sum_{i=1}^{N^{(1)}}\beta_k^{(1)}W_{ik}^{(1)2} + \sum_{j=1}^{N^{(1)}}\beta_k^{(1)}H_{kj}^{(1)2}\\ &+\sum_{k=1}^{K}(\beta_k^{(1)}b^{(1)} - (a^{(1)}-1)\log\beta_k^{(1)})\\ &-2N\log\beta_k^{(1)}) + \mathcal{C},\end{aligned} \tag{1}$$

where $\mathcal{C}$ denotes a constant.

For the case of snapshot t ($2 \leq t \leq T$), similiarly, the observed adjacency matrix $A^{(t)}$ is influenced by an unobserved expectation snapshot network $\hat{A}^{(t)}$, of which element $\hat{A}_{ij}^{(t)}$ denotes the expected expected link weight that take place between i and j at snapshot t. The expectation snapshot network can be composed of a basis matrix $W^{(t)} \in R_+^{N^{(t)}\times K^{(t)}}$ and a community membership matrix $H^{(t)} \in R_+^{K^{(t)}\times N^{(t)}}$ so that $A_{ij}^{(t)} \approx \hat{A}^{(t)} = W^{(t)}H^{(t)}$, where $H_{kj}^{(t)}$ captures the propensity that node j belonging to community k and $K^{(t)}$ is the unknown number of communities. Differently, we consider the historical structure information into the model of current snapshot according to the core idea of evolutionary clustering. In addition, we introduce an evolution matrix $Z^{(t)}$ to model the evolution behaviors of communities synchronously. The element $Z_{lk}^{(t)}$ denotes the propensity that nodes of community l of snapshot $t-1$ transfer into community k of snapshot t. Here, we think the current community membership $H^{(t)}$ is evolved from $H^{(t-1)}$ by modeling some evolution behaviors $Z^{(t)}$, and introduce a penalty term to force that $H^{(t)} \approx \hat{H}^{(t)} = Z^{(t)T}H^{(t-1)}$. In detail, we assume $H_{jk}^{(t)}$ is drawn from a Poisson distribution with rate $\hat{H}_{kj}^{(t)} = \sum_{l=1}^{K^{(t-1)}} Z_{lk}^{(t)} H_{lj}^{(t-1)}$. And $W_{ik}^{(t)}$ and $Z_{lk}^{(t)}$ are both drawn from a half normal distribution with scale parameters $\boldsymbol{\beta}^{(t)} = \{\beta_k^{(t)}\}$. And $\boldsymbol{\beta}^{(t)}$ is drawn from Gamma distribution with two hyper-parameters $a^{(t)}$ and $b^{(t)}$. According to the graphical model in Fig.1, the joint distribution at snapshot t can be represented as follows:

$$\begin{aligned}P(A^{(t)},&H^{(t-1)}, W^{(t)}, H^{(t)}, Z^{(t)}, \boldsymbol{\beta}^{(t)}, \alpha)\\ &= P(A^{(t)}|W^{(t)}, H^{(t)}) \cdot P(H^{(t)}|H^{(t-1)}, Z^{(t)}, \alpha)\\ &\cdot P(Z^{(t)}|\beta^{(t)}) \cdot P(W^{(t)}|\beta^{(t)}) \cdot P(\beta^{(t)}),\end{aligned} \tag{2}$$

where α is a balance parameter. The corresponding posterior is

$$\begin{aligned}P(W^{(t)},&H^{(t)}, Z^{(t)}, \boldsymbol{\beta}^{(t)}|A^{(t)}, H^{(t-1)}, \alpha)\\ &= \frac{P(A^{(t)}, H^{(t-1)}, W^{(t)}, H^{(t)}, Z^{(t)}, \boldsymbol{\beta}^{(t)}, \alpha)}{P(A^{(t)}, H^{(t-1)}, \alpha)}.\end{aligned} \tag{3}$$

The task of minimizing the negative log posterior, which is equivalent to maximize the posterior, can be regarded as the objective function of snapshot t, and its specific form is as follows:

$$\begin{aligned}L^{(t)} = &-\log P(A^{(t)}|\hat{A}^{(t)}) - \log P(H^{(t)}|H^{(t-1)}, Z^{(t)}, \alpha)\\ &-\log P(Z^{(t)}|\beta^{t}) - \log P(\boldsymbol{\beta}^{(t)}).\end{aligned} \tag{4}$$

We assume that $A_{ij}^{(t)}$ is drawn from a Poisson distribution with rate $\hat{A}^{(t)} = \sum_{k=1}^{K^{(t)}} W_{ik}^{(t)} H_{kj}^{(t)}$, $H_{jk}^{(t)}$ is drawn from a Poisson distribution with rate $\hat{H}_{kj}^{(t)} = \sum_{l=1}^{K^{(t-1)}} Z_{lk}^{(t)} H_{lj}^{(t-1)}$, $W_{ik}^{(t)}$ and $Z_{lk}^{(t)}$ are both drawn from a half normal distribution with scale parameters $\boldsymbol{\beta}^{(t)} = \{\beta_k^{(t)}\}$, and $\boldsymbol{\beta}^{(t)}$ is drawn from Gamma distribution with $a^{(t)}$, and $b^{(t)}$, then $L^{(t)}$ can be rewritten as

$$\begin{aligned}L^{(t)} = &\sum_{i=1}^{N^{(t)}}\sum_{j=1}^{N^{(t)}}(A_{ij}^{(t)}\log\frac{A_{ij}^{(t)}}{\sum_{k=1}^{K_{initial}} W_{ik}^{(t)}H_{kj}^{(t)}}\\ &+\sum_{k=1}^{K_{initial}} W_{ik}^{(t)}H_{kj}^{(t)} - A_{ij}^{(t)})\\ &+\alpha\sum_{i=1}^{N^{(t)}}\sum_{k=1}^{K_{initial}}(H_{ki}^{(t)}\log\frac{H_{ki}^{(t)}}{\sum_{l=1}^{K^{(t-1)}} H_{li}^{(t-1)}Z_{lk}^{(t)}}\\ &+\sum_{l=1}^{K^{(t-1)}} H_{li}^{(t-1)}Z_{lk}^{(t)} - H_{ki}^{(t)})\\ &+\sum_{l=1}^{K^{(t-1)}}\sum_{k=1}^{K_{initial}}(\frac{1}{2}\beta_k^{(t)}Z_{lk}^{(t)2}) - \frac{K^{((t-1))}}{2}\log\beta_k^{(t)}\\ &+\sum_{i=1}^{N^{(t)}}\sum_{k=1}^{K_{initial}}(\frac{1}{2}\beta_k^{(t)}W_{ik}^{(t)2}) - \frac{N^{(t)}}{2}\log\beta_k^{(t)}\\ &+\sum_{k=1}^{K_{initial}}(\beta_k^{(t)}b^{(t)} - (a^{(t)}-1)\log\beta_k^{(t)}) + \mathcal{C},\end{aligned} \tag{5}$$

where $\mathcal{C}$ is a constant.

C. Updating Rules

For snapshot 1, the updating rules of the objective function $L^{(1)}$ are the same as Ref. [7], in detail,

$$H^{(1)} \leftarrow (\frac{H^{(1)}}{W^{(1)T}\mathbf{1} + diag(\beta^{(1)})H^{(1)}}) \odot (W^{(1)T}\frac{A^{(1)}}{W^{(1)}H^{(1)}}), \tag{6}$$

$$W^{(1)} \leftarrow (\frac{W^{(1)}}{\mathbf{1}H^{(1)T} + W^{(1)}diag(\beta^{(1)})}) \odot (\frac{A^{(1)}}{W^{(1)}H^{(1)}}H^{(1)T}), \tag{7}$$

$$\beta_k^{(1)} \leftarrow \frac{N^{(1)} + a^{(1)} - 1}{\frac{1}{2}(\sum_i W_{ik}^{(1)2} + \sum_j H_{kj}^{(1)2}) + b^{(1)}}. \tag{8}$$

Similarly for snapshot t ($t \in [2, T]$), we optimize Eq.5 for $W^{(t)}$, $H^{(t)}$, $Z^{(t)}$ and $\boldsymbol{\beta}^{(t)}$ with a gradient descent algorithm, and the updating rules is as follows:

$$H^{(t)} \leftarrow (\frac{H^{(t)}}{W^{(t)T}\mathbf{1} + \alpha}) \odot (W^{(t)T}\frac{A^{(t)}}{W^{(t)}H^{(t)T}} + \alpha\frac{Z^{(t)T}H^{(t-1)}}{H^{(t)}}) \tag{9}$$

$$W^{(t)} \leftarrow (\frac{W^{(t)}}{\mathbf{1}H^{(t)T} + W^{(t)}diag(\beta^{(t)})}) \odot (\frac{A^{(t)}}{W^{(t)}H^{(t)}}H^{(t)T}) \tag{10}$$

$$Z_{lk}^{(t)} \leftarrow (\frac{Z^{(t)}}{Z^{(t)}diag(\boldsymbol{\beta}^{(t)} + \alpha H^{(t-1)}\mathbf{1}}) \odot (\alpha H^{(t-1)}(\frac{H^{(t)}}{Z^{(t)T}H^{(t-1)}})^T) \tag{11}$$

$$\beta_k^{(t)} \leftarrow \frac{K^{(t-1)} + N^{(t)} + 2a^{(t)} - 2}{\sum_l^{K^{(t-1)}} Z_{lk}^{(t)2} + \sum_{l=1}^{N^{(t)}} W_{ik}^{(t)2} + 2b^{(t)}} \tag{12}$$

We update iteratively $W^{(t)}, H^{(t)}, Z^{(t)}$ and $\boldsymbol{\beta}^{(t)}$ according to the above rules until converges. We determine automatically the most appropriate number of communities of each snapshot with a statistical model selection method. In detail, we set a large value (e.g., $K_{initial}^{(t)} = N^{(t)}/4$) as the initial number of communities. After parameter optimization, we shrink $W^{(t)}, H^{(t)}, Z^{(t)}$ to $W^{(t)*}$, $H^{(t)*}$, and $Z^{(t)*}$ by remove the irrelevant rows or columns of which sum is zero or very close to zero. The pseudocode of the solving algorithm of EvoBNMF is presented in Algorithm 1. The returned community label vectors $C^{(t)}(t \in [1, T])$ are the results of temporal community detection. And the returned evolution matrices $Z^{(t)*}$ are the results of quantifying evolution behaviors.

According to the Algorithm 1, the iteratively updating of $H^{(t)}$ is most time-consuming. The time complexity of each iterative is $O(N^2K_{initial} + NK_{initial}^2)$. Here, we set the the average number of iterations as ρ, and the whole time complexity is about $O(\rho T(N^2K_{initial} + NK_{initial}^2))$. As we know, dynamic networks are ususally very sparse in real cases. Then N^2 can be replaced with the average number of edges $\hat{E}$ approximatively at each snapshot. In addition, $K_{initial}$ can be ignored as it is usually much less than N. Naturally, the time complexity of the optimization algorithm of EvoBNMF can degrade to $O(\rho T(\hat{E} + N))$.

Algorithm 1 EvoBNMF

```
Initialize $W^{(t)}$, $H^{(t)}$, $Z^{(t)}$, $\boldsymbol{\beta}^{(t)}$, where $t \in [1, T]$;
while not converge do
    Update $W^{(1)}$, $H^{(1)}$, $\boldsymbol{\beta}^{(1)}$ according to Eq.6 $\sim$ 8;
for $t \in [2, T]$ do
    while not converge do
        $W^{(t)}$, $H^{(t)}$, $Z^{(t)}$, $\boldsymbol{\beta}^{(t)}$ according to Eq.9 $\sim$ 12;
for $t \in [1, T]$ do
    shrink $W^{(t)}$, $H^{(t)}$, and $Z^{(t)}$ to $W^{(t)*}$, $H^{(t)*}$, and $Z^{(t)*}$;

    $C_i^{(t)} = \arg\max_k (H_{ik}^{(t)*})$;
return $C^{(t)}$, $Z^{(t)*}$.
```

III. Experiments

A. Settings

1) Datasets: We test the performance of our EvoBNMF on the eight dynamic networks. Four networks are generated according to SYN-FIX [8], and the other four networks are from real-world KIT-mail [9]. We show their statistical information including the number of snapshots T, the average number of nodes $|\bar{V}|$, the average number of edges $|\bar{E}|$ and the average number of $\bar{K}$ in Tab.II.

Table II
STATISTICAL INFORMATION OF DYNAMIC NETWORKS

Datasets	Networks	T	$\lvert V\rvert$	$\lvert E\rvert$	K
SYN-FIX [8]	Net.1	10	128	2048	4
	Net.2	10	128	2374	4
	Net.3	10	128	1977	4
	Net.4	10	128	2419	4
LFR [10]	Net.5	10	128	2048	4
	Net.6	10	128	2374	4
	Net.7	10	128	1977	4
	Net.8	10	128	2419	4
KIT-mail [9]	Net.9	24	138	29481	23
	Net.10	16	170	29963	25
	Net.11	12	195	29788	25
	Net.12	8	231	27883	27

2) Evaluation Metrics: The performance of community detection is evaluated with two widely-used indexes: the Normalized Mutual Information (NMI) and Error Rate (ER)

[8]. And the accuracy of the autonomous determination of the number of communities is evaluated with Accuracy in the number of communities (KA) [11].

B. Experimental Results

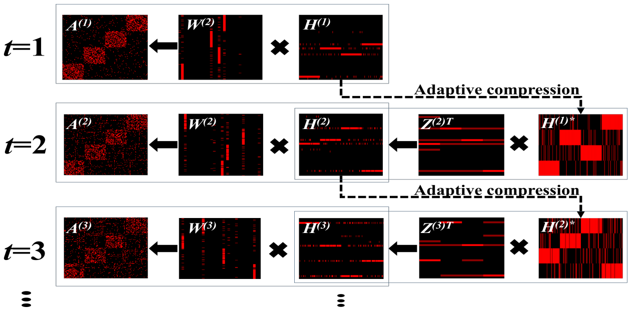

Figure 2. An illustration of EvoBNMF on Network 1.

1) Illustrative example: To clarify the working principle of EvoBNMF, we take an illustrative example on the results of Net.1 in Fig.2. Due to space constraints, we just show the results of snapshot $1-3$. At snapshot 1, the learned matrices $W^{(1)}$ and $H^{(1)}$ are decomposed from the observed $A^{(1)}$. Obviously, there are just four columns of $W^{(1)}$ and four rows of $H^{(1)}$ have high value. And the number of rows of $H^{(1)*}$ is the targeted number of communities after the adaptive compression of the rows with a low value. Then the compressed $H^{(1)*}$ and the observed $A^{(2)}$ are both the input of model at snapshot 2. For snapshot $t(t \geq 2)$, the matrices $W^{(t)}$ and $H^{(t)}$ are decomposed from the observed $A^{(t)}$ and the matrix $H^{(t-1)*}$ and $Z^{(t)}$ are decomposed from the $H^{(t)}$ synchronously in a unified model. It's worth noting that we can obtain the evolution matrices $Z^{(t)*}$ after the adaptive compression of the rows of $Z^{(t)}$. And the evolution matrices correspond to quantitative results of evolution behavior of communities.

2) Overall Performance: To investigate the effectiveness, we compare the performance of our proposed EvoBNMF with four state-of-the-art methods, including Dyluvain [3], PisCES [12], DYNMO [8], ESPRA [13]. We set the hyper-parameter $\alpha = 0.2$, $a = 8$, and $b = 5$ in experiments.

Tab. 4 shows the results over NMI, ER, and KA of the five methods on Net.$1-4$ of SYN-FIX. The best results are bolded out and demonstrate that Dyluvain and our proposed EvoBNMF are comparable in performance and both better than others. The reason is that Dyluvain optimizes the temporal modularity with a greedy heuristic method and is suitable to the synthetic data SYN-FIX.

Furthemore, we shows the results over NMI, ER, and KA of the five methods on network $5-8$ from top to bottom in Fig. 3 respectively. These results are the average results of ten repetitions including the corresponding variance bar. In addition, the x-axis is the snapshot label t, the y-axis is NMI or ER or KA values. From all the subfigures, DyLouvain and EvoBNMF have higher NMI and ER values, and DYNMO and EvoBNMF have higher KA values. This is a strong indication that the proposed EvoBNMF has superior performance not only on temporal community detection but also on autonomous determination of the number of communities.

Table III
THE COMPARISON RESULTS OF FIVE METHODS ON NET.$1-4$

Net.	Index	Dyluvain	PisCES	DYNMO	ESPRA	EvoBNMF
1	NMI	**1**	0.98	0.59	0.97	0.99
	ER	**0**	229.94	3256.91	138	61.56
	KA	**1**	**1**	0.38375	**1**	**1**
2	NMI	**1**	0.97	0.56	0.39	**1**
	ER	**0**	244.34	3340.43	4125	**0**
	KA	**1**	**1**	0.375	0.5625	**1**
3	NMI	0.44	0.30	0.23	0.28	**0.45**
	ER	3630.99	7615.98	4436.61	4467.8	**3576.38**
	KA	0.54	**0.75**	0.069	0.7125	0.69
4	NMI	**0.97**	0.92	0.41	0.29	0.92
	ER	**128.96**	559.14	4024.73	4942.2	500.3
	KA	**1**	**1**	0.31	0.61	0.99

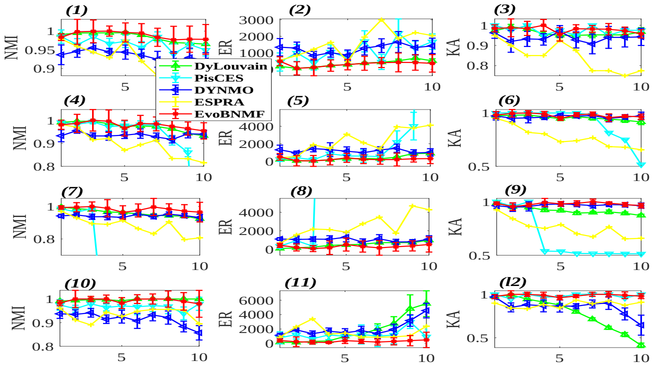

Figure 3. The comparison result of five methods on Net.$5-8$.

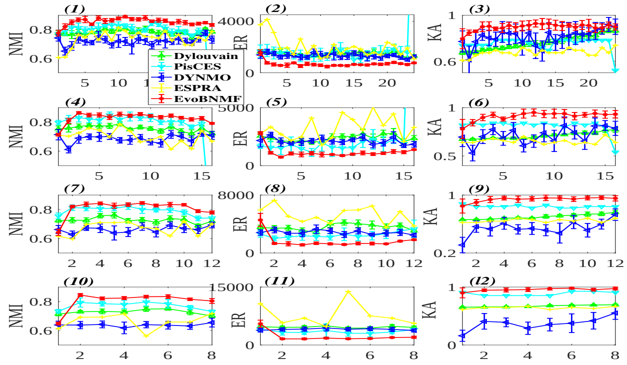

Figure 4. The comparison result of five methods on Net.$9-12$.

Similarly, the subfigures in Fig.4 show the results over

NMI, ER, and KA of the five methods on network $9-12$ from top to bottom respectively. From all the subfigures, we found that the results over NMI, ER and KA of EvoBNMF has the highest accuracy in most cases, but not at the first snapshot. The main reason is that there is has no historical structure information for the first snapshot, and EvoBNMF degenerates to BNMF. In additon, there is significantly improved accuracy from snapshot 1 to snapshot 2, which fully demonstrates the effectiveness of EvoBNMF.

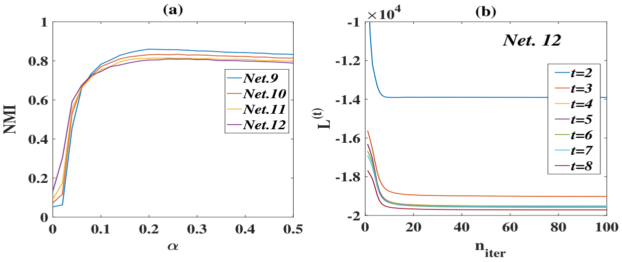

Figure 5. (a) Parameter analysis of the balance parameter α on Net. $9-12$; (b) the convergence analysis of $L^{(t)}$ on Net.12.

3) Parameter Sensitivity and Algorithm convergence: we test the sensitivity of the balance parameter α of EvoBNMF on Net.$9-12$ over NMI by ranging $\alpha \in [0, 0.5]$ with steplength of 0.02. As shown in Fig. 5(a), the performance of EvoBNMF is not sensitive when parameter $\alpha \geq 0.1$, of which is best on about 0.2.

In addition, we verify the convergence of EvoBNMF on Net. 12 with $\alpha = 0.2$. Fig.5 (b) shows the convergence of $L^{(t)}$ at snapshot $2-5$ of Net.12. We find that the value of $L^{(t)}$ always tends to converge when the times of iterations n_{iter} are no more 50, which demonstrates the convergence rate is relatively fast.

IV. Conclusion

We propose EvoBNMF modeling temporal community structure with evolution characteristics for boosting community detection and tracing the corresponding evolution behaviors synchronously in dynamic networks. Meanwhile, we develop a gradient descent algorithm to optimize our model. The numbers of communities could be determined automatically by shrinking the evolution behavior in EvoBNMF. Finally, experimental results on synthetic and real-world networks demonstrate the effectiveness of EvoBNMF. In the future, we will do some predictive tasks of dynamic networks (e.g., links or community structures prediction).

Acknowledgment

This work was supported by the National Key R & D Program of China 2018YFC0809800, the Applied Basic Research Project of Qinghai Province (2018-ZJ-707) and the National Public Complaints and Proposals Administration(2019G0704).

References

[1] G. Rossetti and R. Cazabet, "Community discovery in dynamic networks: a survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 2, p. 35, 2018.

[2] N. Dakiche, F. B.-S. Tayeb, Y. Slimani, and K. Benatchba, "Tracking community evolution in social networks: A survey," *Information Processing & Management*, vol. 56, no. 3, pp. 1084–1102, 2019.

[3] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *science*, vol. 328, no. 5980, pp. 876–878, 2010.

[4] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary clustering," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2006, pp. 554–560.

[5] N. İlhan and Ş. G. Öğüdücü, "Feature identification for predicting community evolution in dynamic social networks," *Engineering Applications of Artificial Intelligence*, vol. 55, pp. 202–218, 2016.

[6] W. Yu, W. Wang, P. Jiao, and X. Li, "Evolutionary clustering via graph regularized nonnegative matrix factorization for exploring temporal networks," *Knowledge-Based Systems*, vol. 167, pp. 1–10, 2019.

[7] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon, "Overlapping community detection using bayesian non-negative matrix factorization," *Physical Review E*, vol. 83, no. 6, p. 066114, 2011.

[8] F. Folino and C. Pizzuti, "An evolutionary multiobjective approach for community discovery in dynamic networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1838–1852, 2014.

[9] (2011) Karlsruhe institue of technology dynamic network of email communication at department of informatics at karlsruhe institue of technology (kit). [Online]. Available: http://i11www.iti.uni-karlsruhe.de/en/projects/spp1307/emaildata

[10] D. Greene, D. Doyle, and P. Cunningham, "Tracking the evolution of communities in dynamic social networks," in *2010 International Conference on Advances in Social Networks Analysis and Mining*. IEEE, 2010, pp. 176–183.

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

[12] F. Liu, D. Choi, L. Xie, and K. Roeder, "Global spectral clustering in dynamic networks," *Proceedings of the National Academy of Sciences*, vol. 115, no. 5, pp. 927–932, 2018.

[13] P. Wang, L. Gao, and X. Ma, "Dynamic community detection based on network structural perturbation and topological similarity," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2017, no. 1, p. 013401, 2017.