Generative Evolutionary Anomaly Detection in Dynamic Networks

Pengfei Jiao, Tianpeng Li, Yingjie Xie, Yinghui Wang, Wenjun Wang,
Dongxiao He, and Huaming Wu

Abstract—Anomaly detection in dynamic networks aims to find network elements (e.g., nodes, edges, subgraphs, change points) with significantly different behaviors from the vast majority, it can also devote to community detection and evolution and prediction tasks. Most existing methods focus on one specific task, that is, only detect anomalies of one type of element isolated, so they lose the ability to model the correlation and driving mechanism between different abnormal behavior. Considering that the anomaly detection of one type of element is helpful to other types of elements, i.e., the temporal evolution hidden the dynamic networks are driven by indivisible behavior patterns. So in this paper, we propose a unified Generation model to analyze the dynamic network for Exploring the Abnormal Behaviors of different Scales (GEABS). It can model the relation and catch different levels (node, community and network) of anomaly with a joint statistical network model and detect the community structure and its evolution. Specifically, we denote the parameters of node popularity, community membership to generate the dynamic network with stochastic block model (SBM), we also describe the varying of node and community by dynamic process. With a well-designed generative mechanism, it can detect the change point on network level, temporal evolution on community level and abnormal behavior on node level synchronously, besides, it also detects the community structure effectively. We also propose an effective optimization algorithm with variational inference. Experimental results show that the GEABS achieves better performance on abnormal behavior and community structure compared with baselines.

Index Terms—Dynamic network, abnormal behavior, different levels, generation model, stochastic block model

◆

1 INTRODUCTION

ANOMALY detection is an important task in data mining and pattern recognition [1], [2]. Its purpose is to find cases that hide rare patterns in a given data set that do not conform to the expected normal behavior. Anomaly detection attracts much research and has been applied to many fields, such as fraud detection [3] and abnormal behavioral detection [4]. So a variety of methods have been proposed in the past years. For that complex network can be used in modeling a myriad of real-world systems, ranging from international air transportation [5] to team collaboration networks [6], from social networks [7] to citation networks [8], recent years, network anomaly detection has also been received much attention and has become one of the key issues in current research. Similar to anomaly detection, but beyond it, anomaly detection in complex networks usually refers to find nodes whose patterns significantly deviate from the vast majority of the network [9], [10] or that trigger the significant structural changes of the network at different levels (i.e., micro level, mesoscopic level and macro level) [11].

Since the network structure itself evolves over time (dynamic or temporal network) [12], such as the addition and disappearance of nodes and varying of edges which also drive the dynamic behavior of communities. So it is difficult to determine whether changes in a dynamic network are normal or abnormal. The causes of network anomalies are also complex, and there are many forms of anomalies denoted, including abnormal nodes, abnormal edges, abnormal subgraphs, detection of change points [11]. At the same time, anomaly detection in dynamic networks helps to better understand the evolution status of the network, assess the anomaly degree of the network and its impact, and formulate effective intervention measures to deal with potential crises in the network. For example, it can improve the accuracy of community detection by properly using anomaly detection to find the non-smooth evolution time point, which is caused by the evolution of the network. It can also be used in anomaly gene identification biological systems, climate prediction and financial markets.

Generally, most existing anomaly detection methods are usually based on a specific perspective, i.e., tracking the evolution of dynamic networks at different times at the macro, micro, and mesoscopic levels, respectively, ignoring the mutual influence of anomalies at the three levels. For example, [13], [14] focus on detecting anomalies at the micro level (i.e., abnormal nodes or edges), [15], [16], [17] focus on anomaly detection of various neural networks to modify the

- Pengfei Jiao is with the School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China. E-mail: cspjiao@gmail.com.
- Tianpeng Li, Yingjie Xie, Yinghui Wang, Wenjun Wang, and Dongxiao He are with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China. E-mail: {ltpnimeia, yjxie, wangyinghui, wjwang, hedongxiao}@tju.edu.cn.
- Huaming Wu is with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China. E-mail: whming@tju.edu.cn.
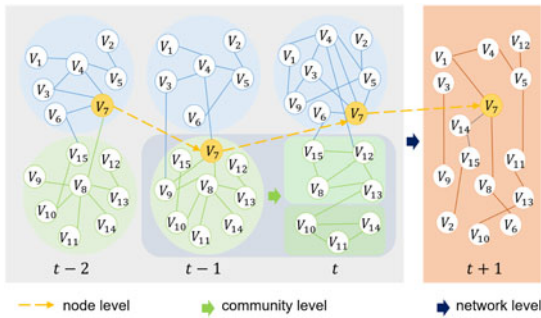
Fig. 1. A toy example to show the temporal evolution with different scale abnormal behaviors and their relations in dynamic networks.

convergence performance. and [18], [19] focus on anomaly detection at the macro level (i.e., change points). In fact, in the process of network evolution, it is highly possible that multiple types of abnormalities have occurred. And there is a correlation between anomalies at different levels, and anomaly detection at one level is also helpful for anomaly detection at other levels. Take anomaly detection in the financial market as an example. In a financial transaction network, nodes are accounts, and edges are transactions between accounts. Then the abnormal events at the micro level can be nodes that frequently trade with different communities. In real life, this account may be a normal account, but it may also be an account used for fraud. An abnormal event at the mesoscopic level can be a group of accounts that frequently trade with each other before suddenly splitting into two or more groups. This group of accounts, in real life, maybe all belonging to a large company, then the split event may indicate that this company has been reorganized. The abnormal event at the macro level is that the entire market has experienced great fluctuation, which is reflected in the financial market network as a point of network change. Furthermore, the accumulation of abnormal accounts can cause drastic changes in account groups, and the big events of the whole market may affect all the account groups and accounts(such as the financial crisis). As shown in Fig. 1, the state of node v_7 in the network is constantly changing with time, so it can be regarded as an abnormal point in the evolution of the entire network. In other words, its anomaly causes the entire network anomaly. However, the anomaly of v_7 can also be seen as the result of an anomaly at the mesoscopic level (i.e., division and merger of communities). This results in the coupling of micro-level (abnormal node) and mesoscopic-level (abnormal subgraph) anomalies in the process of network evolution. Only considering a single type of abnormality is not enough to support the complete mining of the law of network evolution. That is, the abnormalities of the network at the macro, micro, and mesoscopic levels are closely related to each other and are driven by deeper factors. The anomaly is only the appearance, and the essence is unknown.

Only a few heuristic methods consider the correlation between anomalies at different levels, such as [18] and [19] detect community structures anomaly and change points simultaneously. However, they only consider some but not all levels of anomalies, and some need to manually define features. Besides, these methods do not have a unified framework, they realize different levels of anomaly

detection step by step, which cannot model the coupling relationship between anomalies to achieve mutual enhancement. To catch the correlation between anomalies at different levels and better detect the anomaly of node, community, change points and the entire network, we use a unified Generation model to analyze the dynamic network for Exploring the Abnormal Behaviors of different Scales (GEABS) due to the poor interpretability of anomaly detection methods based on network characteristics [20], [21], [22]. We use the characteristic that the generative model has good interpretability to explore the essence of this abnormality in this paper. Specifically, we denote the parameters of node popularity, community membership to generate the dynamic network with stochastic block model (SBM), and accurately identify network evolution abnormalities by whether the parameter changes exceed the threshold calculated by a defined operator. In addition to detecting anomalies, our model can also detect the community structure. We optimize our method by variational inference. The main contributions of this paper can be categorized as follows:

- We propose an anomaly detection method based on a generative model, which takes into account the interaction of anomalies at different levels, and captures all types of anomalies that may occur during the evolution of the network.
- We realize the correlation analysis of network macroscopic, mesoscopic, and micro-scale abnormalities by introducing network parameters and transition matrix based on Stochastic Block Model (SBM).
- We use variational inference to achieve efficient optimization. Our model has good versatility, can be easily applied to various generative models, and has good interpretability.
- Experimental results demonstrate that the proposed model can significantly improve the performance of network anomaly detection compared with the state-of-the-art on both synthetic datasets and real-world networks.

The rest of this paper proceeds as follows: Section 2 presents the related work on anomaly detection and network anomaly detection. Section 3 formally describes our proposed method and notations we used. Related inference is represented in Section 4. Then we validate our approach by analyzing extensive experiments in Section 5 and summarize our paper in Section 6.

2 RELATED WORK

Anomaly detection in dynamic networks usually refers to find nodes whose patterns significantly deviate from the vast majority of nodes or that trigger the significant structural changes of the network at different levels. Therefore, we roughly divide existing methods into three categories: micro, mesoscopic and macro level in this paper.

Micro Level. It is mainly to detect abnormal vertices and abnormal edges in the network [13], [23], [24], [25]. Hassanzadeh *et al.* [13] proposed a semi-supervised method for detecting abnormal nodes in online social networks by extracting features using an expectation-maximizing clustering algorithm to classify vertices according to the initial

abnormity score of vertices, and using fuzzy logic of membership functions to define the degree of abnormity. Heard *et al.* [14] used the Bayesian discrete-time counting process to model the number of edges between vertices, by learning the distribution of edges between vertices, to calculate the probability P-value of the newly observed edge, and use this value to mark abnormal edges. Sun *et al.* [26] utilized a compact matrix decomposition to compute sparse low-rank approximations to realize the exploration of abnormal edges in the network. Zheng *et al.* [23] proposed a general end-to-end anomalous edge detection framework using an extended temporal GCN with an attention model. Charu *et al.* [24] used a structural connectivity model in order to define outliers in graph streams and Ranshous *et al.* [25] described a high-level model for outlier detection based on global and local structural properties of a stream.

Mesoscopic Level. It is mainly to detect abnormal subgraphs in the network [27], [28], [29], [30], [31], [32], [33], [34]. Mongiovi *et al.* [29] hypothesized that the gradual change of subgraphs in consecutive time steps may cause the network to shrink or grow between consecutive snapshots and using the edge distance as a distance metric to determine the distance between two adjacent subgraphs. Miller *et al.* [30] detected anomalously sparse (rather than anomalously dense) subgraphs by embedding the signal involves subtracting edges. Shin *et al.* [27] proposed a disk-based dense-block detection method, which also can be run in a distributed manner across multiple machines and proposed a flexible framework for finding dense blocks in tensors [28]. Eswaran *et al.* [31] proposed a randomized sketching-based approach called SpotLight, which guarantees that an anomalous graph is mapped 'far' away from 'normal' instances in the sketch space with a high probability for the appropriate choice of parameters.

Since the evolution of the network is non-smooth, some studies of abnormal points detection focus on the detection of community evolution [35], [36], [37], [38] and regard the changes in the community as an anomaly. Baingana *et al.* [38] advocated a novel approach for jointly tracking communities, while detecting such anomalous nodes in time-varying networks. Bhat *et al.* [35] proposed a unified framework that adapts a preliminary community structure towards dynamic changes in social networks using a novel density-based approach for detecting overlapping community structures. A new method for the group evolution discovery was proposed in [36]. Ma *et al.* [37] proposed two evolutionary nonnegative matrix factorization frameworks for detecting dynamic communities.

Macro Level. It is mainly to detect change points in the network [18], [19], [24], [39], [40], [41], [42]. Peel *et al.* [40] proposed the GHRG model, which uses a hierarchical random graph to simulate the nested community structure in the dynamic network structure. This method uses the concept of a fixed-length sliding window and a generalized likelihood ratio to evaluate whether the network structure has changed significantly within the sliding window time. LAD [43] calculates the Singular Value Decomposition (SVD) of the graph Laplacian operator to obtain a low-dimensional graph representation. It uses two sliding windows to explicitly model short-term and long-term dependencies to capture sudden changes and gradual changes in

the dynamic network. SCOUT [18] works by finding the set of change points and community structures that minimizes an objective criterion derived from the Akaike Information Criterion or Bayesian Information Criterion [44]. Cheung *et al.* [19] proposed a novel methodology to detect both community structures and change points simultaneously based on a model selection framework, in which the Minimum Description Length (MDL) is utilized as minimizing objective criterion.

These methods only focus on one or two-level anomalies, most consider abnormal community detection and change point detection simultaneously. So in this paper, we propose a unified framework to model anomalies at these three different levels and catch their relevance.

3 THE GENERATIVE MODEL

In this section, we will introduce some preliminaries and the detailed generative model for dynamic networks.

3.1 Preliminaries

Given a dynamic network $G = \{G^{(1)}, G^{(2)}, \ldots, G^{(T)}\}$, each $G^{(t)}(1 \leq t \leq T)$ is the snapshot network at t. We denote the $N^{(t)}$, $K^{(t)}$ and $W^{(t)}$ as the number of nodes, communities and the similarity matrix of snapshot network $G^{(t)}$. Usually, we could assume that $N^{(t)} = N$ is constant with t and $W^{(t)}$ and $K^{(t)}$ are varying with time. If the network is undirected and binary, we denote $W_{ij}^{(t)} = 1$ if nodes V_i and V_j share a link at snapshot t and $W_{ij}^{(t)} = 0$ otherwise. Here we assume the dynamic network satisfies this type, however, it can be easily extended to weighted or directed networks.

For the dynamic network G, we denote the community structure as $Z = \{Z^{(1)}, Z^{(2)}, \ldots, Z^{(T)}\}$. Here $Z^{(t)} \in \{0, 1\}^{K_t \times K_t}$, and $Z_{ik}^{(t)} = 1$ means node V_i belonging community k and K_t is the number of communities at snapshot t. In our model, we only consider the non-overlapping community, i.e., $\sum_{k=1}^{K_t} Z_{ik}^{(t)} = 1$. Furthermore, we also denote $z_i^{(t)} = k$ instead of $Z_{ik}^{(t)} = 1$ for convenience.

Based on the above, we are committed to discovering abnormal patterns in the network. We formalize it as following.

- *Macro level (network anomaly)*: it is usually same to change point, i.e., we find a set $\sqcup = \{t : |f(G^{(t-1)}) - f(G^{(t)})| > \epsilon\}$, f is a function on snapshot network and ϵ is a predefined threshold.
- *Mesoscopic level (community anomaly)*: if community structure $Z^{(t-1)}$ and $Z^{(t-1)}$ have a significant difference, it means there are community events, e.g., community growth and contraction, merge and split.
- *Micro level (node anomaly)*: Based on the community membership $Z = \{Z^{(1)}, Z^{(2)}, \ldots, Z^{(T)}\}$, if one node changes community affiliation frequently, it should be abnormal in the dynamic network.

Here, we call these questions an evolutionary anomaly in a dynamic network. The first one is easy to analyze and evaluate, however, the last two questions are difficult to quantify. In our GEABS model, we will analyze the network anomaly detection from the perspective of the generative model with community detection. A list of major notations involved in this paper is summarized in Table 1.

Symbol	Description
G, $G^{(t)}$	The dynamic network and its tth snapshot
$N^{(t)}$, $K^{(t)}$	The number of nodes and communities of $G^{(t)}$
$W^{(t)}$, $W_{ij}^{(t)}$	The similarity matrix and its element of $G^{(t)}$
λ	The exponential prior parameter of $\delta_i^{(t)}$
π	The multinomial prior parameter of $z_i^{(t)}$
μ	The Dirichlet prior parameter of $A^{(t)}$
B	The association probability matrix of community
$A^{(t)}$	The probability transition matrix from snapshots $t-1$ to t
Z, $Z^{(t)}$	The community structure of G and $G^{(t)}$
$z_i^{(t)}$	The community label of nodes V_i at snapshot t
$\delta^{(t)}$, $\delta_i^{(t)}$	The popularity vector and its ith element at snapshot t
$\phi^{(t)}$, $\phi_j^{(t)}$	The variational parameters of $Z^{(t)}$ and $z_i^{(t)}$
$\bar{\delta}^{(t)}$, $\bar{\delta}_i^{(t)}$	The variational parameters of $\delta^{(t)}$ and $\delta_i^{(t)}$
$\tilde{\mu}^{(t)}$, $\tilde{\mu}_{kl}^{(t)}$	The variational parameters of $A^{(t)}$ and $A_{kl}^{(t)}$

3.2 The Generative Model

Here, we will present the detailed generative process of GEABS model. For a dynamic network, it consists of two parts, i.e., network snapshots $t=1$ and $t \geq 2$.

3.2.1 Generative Process on $G^{(1)}$

At snapshot $t=1$, it is usually modeled as a static network for $G^{(1)}$. As, in the graph model in Fig. 2a the generative process is presented in *Algorithm 1*.

Algorithm 1. Generation Snapshot Network $t=1$

Input: Model parameters λ and π
Output: Similarity matrix $W^{(1)}$
1: Initializing hyperparameters
2:     Sample $\pi \sim Dirichlet(1)$
3:     Sample $\lambda \sim Uniform(0,1)$
4:     Sample $B \sim Beta(1,1)$
5: Sample $Z^{(1)}$ and $\delta^{(1)}$
6: **for** $i = 1, 2, \ldots, N$ **do**
7:     Choose $Z_i^{(1)} \sim Multi(\pi)$
8:     Choose $\delta_i^{(1)} \sim Exp(\lambda)$
9: **end for**
10: Generate the links of $G^{(1)}$
11: **for** $i = 1, 2, \ldots, N-1, j = 2, \ldots, N$ **do**
12:     Choose $W_{ij}^{(1)} \sim Bernoulli(B_{z_i^{(1)} z_j^{(1)}}^{1+\delta_i^{(1)}+\delta_j^{(1)}})$
13: **end for**

This process is similar to the generation of stochastic block model (SBM), $\pi \in (0,1)^{K^{(1)}}$ is the probability vector of the multinomial distribution and $\sum_k \pi_k = 1$. It is also the distribution of the size of communities, i.e., the probability of randomly selecting a node belonging to different communities. $B \in (0,1)^{K^{(1)} \times K^{(1)}}$ is the interaction probability between communities, i.e., the probability of having a link between any two nodes in their respective communities. Then, we can sample the latent variable $Z_i^{(1)}$ for each node from the multinomial distribution with π. Last, based on the SBM, for each node pair $i \neq j$, the $W_{ij}^{(1)}$ is sampled from $Bernoulli(B_{z_i^{(1)} z_j^{(1)}})$. However, it loses the different characteristics of each node,

(a) Graphical model at $t = 1$ (b) Graphical model at $t > 1$
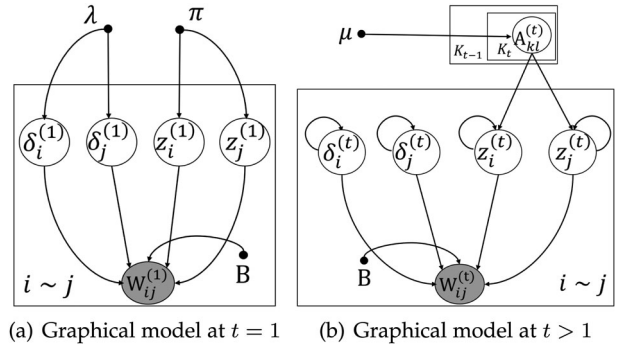
Fig. 2. Graphical model of our proposed generative model.

i.e., it means that nodes of the same community are equivalent. To model the difference of nodes in the network, we design popular parameter $\delta_i^{(1)}$ for node V_i, we treat it as learnable parameters not determined value (such as the degrees) for that it can be used for the anomaly detection in our model, i.e., $W_{ij}^{(1)} \sim Bernoulli \left(B_{z_i^{(1)} z_j^{(1)}}^{1+\delta_i^{(1)}+\delta_j^{(1)}} \right)$. In this way, it can avoid the community of homogeneous results in the network and identify the popularity of nodes in the network, not just its degree. It has also been proven that it can capture the power-law distribution in the network and has an advantage over SBM and degree-corrected stochastic block model.

3.2.2 Generative Process on $G^{(t)}$

For snapshot $t > 2$, we should analyze the evolution across the snapshots from $t-1$ to t. First, the popular parameter $\delta_i^{(t)}$ is depend on $\delta_i^{(t-1)}$, we take the Exponential distribution to describe the relationship, i.e., $\delta_i^{(t)} \sim Exp(\delta_i^{t-1})$. For the community variable, we define a probability transition matrix $A^{(t)}$ to describe the community level evolution, $A_{kl}^{(t)}$ is denoted as the probability of one node transiting from community k at snapshot $t-1$ to community l at t and $\sum_{l=1}^{K^{(t)}} A_{kl}^{(t)}$, for $k = 1, \ldots, K^{(t-1)}$. With the latent variables for network snapshot t, as in Fig. 2b, we can generate the observed structure with *Algorithm 2*.

Here, the probability transition matrix $A^{(t)}$ represents the community level evolution, it could reveal the different behaviors. Considering the relationship between community behavior and anomaly, we will denote its abnormal behavior based on $A^{(t)}$ later. With this generative process, we can model the varying of dynamic networks with node popularity, community structure and its evolution under a unified framework. Then, our GEABS model can reveal the temporal behaviors and the anomaly on different scales.

Here, we give some supplementary remarks about the GEABS model for details.

- The choice of these distributions, e.g., Dirichlet, Beta and multinomial, should satisfy the conjugate distribution and network properties as much as possible.
- Regarding the parameters λ, π and μ, we can use general and conjugate distributions to generate them without additional hyperparameters.
- Changing the number of nodes in the dynamic network, we can add the differential nodes across the

snapshots or take the union of nodes of consecutive snapshots for generating the observed structure.

- We only generate $W_{ij}^{(t)}$ with Bernoulli distribution, which is only suitable for binary networks. For weighted or real-valued networks, Poisson or exponential distributions can be used instead.

Algorithm 2. Generation Snapshot Network $t \geq 2$

Input: Model parameters μ, the popularity $\delta^{(t-1)}$ and community structure $Z^{(t-1)}$
Output: Similarity matrix $W^{(t)}$
 1: Initializing hyperparameters
 2:　　Sample $\mu \sim Dirichlet(1)$
 3:　　Sample $B \sim Beta(1,1)$
 4: Sample $A^{(t)}$
 5: **for** $k = 1, 2, \ldots, K^{(t-1)}$ **do**
 6:　　$A_{k.}^{(t)} \sim Dirichlet(\mu)$
 7: **end for**
 8: Sample $Z^{(t)}$ and $\delta^{(t)}$
 9: **for** $i = 1, 2, \ldots, N$ **do**
10:　　Choose $\delta_i^{(t)} \sim Exp(\delta_i^{(t-1)})$
11:　　Choose $Z_i^{(t)} \sim Multi(A_{Z_i^{(t-1)}}^{(t)})$
12: **end for**
13: Generate the links of $G^{(t)}$
14: **for** $i = 1, 2, \ldots, N-1, j = 2, \ldots, N$ **do**
15:　　Choose $W_{ij}^{(t)} \sim Bernoulli(B_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}})$
16: **end for**

3.3 The Joint Probability of the Model

For our GEABS model, there are two kinds of formalization, online and offline. The former only focuses on the snapshot t and the previous data, the latter deals with all snapshots of the dynamic network. Here, we only give offline formalization to learn the evolution of abnormal patterns.

At snapshot 1, based on the graph model and its generative process, the joint probability distribution of the observable variables $W^{(1)}$ and latent variables $Z^{(1)}$ and $\delta^{(t)}$ is as follows:

$$O_1 = Pr(W^{(1)}, Z^{(1)}, \delta^{(1)} | \pi, \lambda, B)$$
$$= Pr(W^{(1)} | Z^{(1)}, B, \delta^{(1)}) Pr(\delta^{(1)} | \lambda) Pr(Z^{(1)} | \pi). \quad (1)$$

where $Pr(Z^{(1)} | \pi)$ is the probability of community assignment in first snapshot $t-1$, and $Pr(\delta^{(1)} | \lambda)$ is the probability distribution of initial popularity. They can be calculated as

$$Pr(Z^{(1)} | \pi) = \prod_{i=1}^{N} Pr(z_i^{(1)} | \pi) = \prod_{i=1}^{N} \pi_{z_i^{(1)}}, \quad (2)$$

$$Pr(\delta^{(1)} | \lambda) = \prod_{i=1}^{N} Pr(\delta_i^{(1)} | \lambda) = \prod_{i=1}^{N} \lambda e^{-\lambda \delta_i^{(1)}}. \quad (3)$$

Similarity, we can write it at snapshot t as follows:

$$O_t = Pr(W^{(t)}, Z^{(t)}, \delta^{(t)}, A^{(t)} | \mu, Z^{(t-1)}, \delta^{(t-1)}, B)$$
$$= Pr(W^{(t)} | Z^{(t)}, B, \delta^{(t)}) Pr(\delta^{(t)} | \delta^{(t-1)})$$
$$\quad Pr(Z^{(t)} | Z^{(t-1)}, A^{(t)}) Pr(A^{(t)} | \mu). \quad (4)$$

In the case of the probability distribution proposed above, the detailed formalization is as in Eqs. (5)–(7)

$$Pr(W^{(t)} | Z^{(t)}, B, \delta^{(t)}) = \prod_{i \sim j} Pr\left(W_{ij}^{(t)} | z_i^{(t)}, z_j^{(t)}, B, \delta_i^{(t)}, \delta_j^{(t)}\right)$$
$$= \prod_{w_{ij}^{(t)}=1} B_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}} \prod_{w_{ij}^{(t)}=0} \left(1 - B_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}}\right), \quad (5)$$

$$Pr(Z^{(t)} | Z^{(t-1)}, A) = \prod_{i=1}^{N} Pr(z_i^{(t)} | z_i^{(t-1)}, A) = \prod_{i=1}^{N} A_{z_i^{(t-1)} z_i^{(t)}}, \quad (6)$$

$$Pr(\delta^{(t)} | \delta^{(t-1)}) = \prod_{i=1}^{N} Pr(\delta_i^{(t)} | \delta_i^{(t-1)}) = \prod_{i=1}^{N} \delta_i^{(t-1)} e^{-\delta_i^{(t-1)} \delta^{(t)}}, \quad (7)$$

where $Pr(W^{(t)} | Z^{(t)}, B, \delta^{(t)})$ is the conditional probability of $W^{(t)}$, $Pr(\delta^{(t)} | \delta^{(t-1)})$ is the varying probability of node popularity, and $Pr(Z^{(t)} | Z^{(t-1)}, A^{(t)})$ is the transition probability across the snapshots for $t = 2, \ldots, T$.

We choose the exponential distribution to model the varying of $\delta^{(t)}$, i.e., $\delta_i^{(t)} \sim Exp(\delta_i^{(t-1)})$, it is easy to know that $E(\delta_i^{(t)}) = \frac{1}{\delta_i^{(t-1)}}$. With this, if the node popularity has not changed significantly, $\delta_i^{(t)} \approx 1$. However, if it has a significant change and fits the network snapshot well, this means it may be a change point. Besides, with the Dirichlet distribution for the probability transition matrix $A^{(t)}$, the joint probability distribution of the model is given as

$$O = Pr(W, Z, \delta, A | \pi, \lambda, \mu, B) = Q_1 \prod_{t=2}^{T} O_t$$

$$= \prod_{t=1}^{T} \left[\prod_{w_{ij}^{(t)}=1} b_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}} \prod_{w_{ij}^{(t)}=0} \left(1 - b_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}}\right) \right]$$

$$\prod_{t=2}^{T} \prod_{i=1}^{n} A_{z_i^{(t-1)} z_i^{(t)}} \prod_{t=2}^{T} \prod_{k} \frac{\Gamma(\sum_l \mu_{kl})}{\prod_l \Gamma(\mu_{kl})} \prod_l A_{kl}^{\mu_{kl}-1}$$

$$\prod_{t=2}^{T} \prod_{i=1}^{N} \delta_i^{(t-1)} e^{-\delta_i^{(t-1)} \delta_i^{(t)}} \prod_{i=1}^{N} \pi_{z_i^{(1)}} \prod_{i=1}^{N} \lambda e^{-\lambda \delta_i^{(1)}}. \quad (8)$$

3.4 Evolutionary Anomaly Detection

With our GEABS model, if we have learned the variable parameters δ, Z and A, which can characterize the node level activity and popularity, community structure and its evolutionary activity, respectively. To capture the different levels of anomaly, we further define the three levels of anomaly detection.

At the *network level*, it also is called a change point in dynamic networks, its essence is to calculate the difference between two snapshots. By combining the node level popularity δ and the community level transition parameter A and community structure Z, the network level anomaly value at snapshot t is defined as follows:

$$E_{nl}^{(t)} = \sum_{i,k} e^{A_{z_i^{(t)} k}^t (1 - I(z_i^{(t)}, z_i^{(t-1)}))} \|\delta_i^{(t)} - \delta_i^{(t-1)}\|_2^2, \quad (9)$$

where the node popularity, community structure and transition probability are integrated across the snapshots. If the node V_i belongs to the same community, the anomaly value

only depends on the varying of node popularity, otherwise, this value is corrected by the transition probability in $A^{(t)}$. Furthermore, for the set $\sqcup = \{t : |f(G^{(t-1)}) - f(G^{(t)})| > \epsilon\}$, where the function $f = E_{nl}^{(t)}$. Besides, the choice of ϵ is usually replaced by the Top-K values.

At the *community level*, if community structure $Z^{(t-1)}$ and $Z^{(t)}$ have a significant difference, it means there are community events, e.g., community growth and contraction, merge and split. In this paper, we analyze the anomaly behaviors based on the transition probability matrix $A^{(t)}$. Considering that community detection is an unsupervised problem, matching communities on different snapshots is also an interesting issue, with basic mapping way, some important community behaviors, such as merge and split, growth and contraction can be captured with our $A^{(t)}$.

At the *node level*, based on the community membership $Z = \{Z^{(1)}, Z^{(2)}, \ldots, Z^{(T)}\}$, if one node changes community affiliation frequently, it should be abnormal in the dynamic network. So we denote the anomaly value of nodes based on the community structure and as Eq. (10)

$$E_{cl(i)} = \frac{\sum_t^T \delta_i^{(t)}}{T} H(Z_i^{(1):(T)}), \qquad (10)$$

where $H(Z_i^{(1):(T)})$ is the Entropy measure, while $Z_i^{(1):(T)}$ is the total community memberships node i belonging to from $t = 1$ to $t = T$. Therefore, $Z_i^{(1):(T)}$ is a vector representing the whole community membership of node V_i.

4 MODEL INFERENCE

To optimize the joint distribution of Eq. (8), it is required to compute the posterior function given the observed variables and hyper-parameters, which can be written as follows:

$$Pr(Z, \delta, A, \pi, \lambda, B | W, \mu) = \frac{Pr(W, Z, \delta, A, \pi, \lambda, \mu, B)}{Pr(W, \mu)}. \qquad (11)$$

However, it is impossible to calculate Eq. (11) directly, so with the variational inference, a novel variational distribution q is applied to approximate the posterior, which can be defined as follows:

$$q(\phi, \bar{\delta}, \tilde{\mu}) = \prod_t \left[\prod_i q\left(\phi_i^{(t)}\right) \prod_i q\left(\bar{\delta}_i^{(t)}\right) \prod_k \prod_l q\left(\tilde{\mu}_{kl}^{(t)}\right) \right], \qquad (12)$$

where ϕ, $\bar{\delta}$, and $\tilde{\mu}$ are the variational parameters of Z, δ, and A in the joint distribution of Eq. (8). In detailed, it means that $z_i^t \sim Multi(\phi_{kl}^{(t)})$, $\delta_i^{(t)} \sim \mathbf{1}(\delta_i^{(t)})$ and $A_{kl}^{(t)} \sim Dirichlet(\tilde{\mu}_{kl}^{(t)})$. $q(\phi_i^{(t)})$, $q(\delta_i^{(t)})$ and $\tilde{\mu}_{kl}^{(t)}$ are the approximate posterior distributions of $z_i^{(t)}$, $\delta_i^{(t)}$ and $A_{kl}^{(t)}$, respectively.

4.1 Variational Inference E-Step

With the variational inference and the definition in Eq. (12), the log likelihood using Jensen's inequality is bounded as Eq. (13) and it also satisfies $\log Pr(W) = KL(q\|p) + \mathscr{L}(q)$

$$\log Pr(W) \geq E_q(\log Pr(W, Z, \delta, A)) + H(q), \qquad (13)$$

where we ignore the parameters in the log likelihood, and $H(q)$ denotes the Entropy. q and p are the approximate and expected posterior distributions, $\mathscr{L}(q)$ is the ELBO of the

variational inference. Then, minimizing $KL(q\|p)$ can be translated into maximizing $\mathscr{L}(q)$. The log likelihood of complete data and ELBO can be denoted as Eqs. (14) and (15).

$$\log Pr(W, Z, \delta, A | \pi, \lambda, \mu, B)$$

$$= \sum_{t=1}^T \left[\sum_{w_{ij}^{(t)}=1} \left(1 + \delta_i^{(t)} + \delta_j^{(t)} \right) \log b_{z_i^{(t)} z_j^{(t)}} \right.$$

$$+ \sum_{w_{ij}^{(t)}=0} \log \left(1 - b_{z_i^{(t)} z_j^{(t)}}^{1+\delta_i^{(t)}+\delta_j^{(t)}} \right) + \sum_{t=2}^T \sum_i \log A_{z_i^{(t-1)} z_i^{(t)}}^{(t)} \right]$$

$$+ \sum_i \log \pi_{z_i^{(1)}} + N \log \lambda - \lambda \sum_i \delta_i^{(1)}$$

$$+ \sum_{t=2}^T \sum_k \left[\log \Gamma \sum_l \mu_{kl} - \sum_l \log \Gamma(\mu_{kl}) \right.$$

$$\left. + \sum_l (\mu_{kl} - 1) \log A_{kl}^{(t)} \right] + \sum_{t=2}^T \sum_i \left[\log \delta_i^{(t-1)} - \delta_i^{(t-1)} \delta_i^{(t)} \right]. \quad (14)$$

$$\mathscr{L}(Z, \bar{\delta}, \tilde{\mu}; \pi, B, \lambda, \mu)$$

$$= \sum_{t=1}^T \sum_{w_{ij}^{(t)}=1} \left(1 + \bar{\delta}_i^{(t)} + \bar{\delta}_j^{(t)} \right) \sum_k \sum_l \phi_{ik}^{(t)} \phi_{jl}^{(t)} \log B_{kl}$$

$$- \sum_{t=1}^T \sum_{w_{ij}^{(t)}=0} \sum_k \sum_l \phi_{ik}^{(t)} \phi_{jl}^{(t)} B_{kl}^{1+\bar{\delta}_i^{(t)}+\bar{\delta}_j^{(t)}}$$

$$+ \sum_{t=2}^T \sum_i \sum_k \sum_l \phi_{ik}^{(t-1)} \phi_{il}^{(t)} \left[\psi(\tilde{\mu}_{kl}^{(t)}) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t)} \right) \right]$$

$$+ \sum_i \sum_k \phi_{ik}^{(1)} \log \pi_k + N \log \lambda - \lambda \sum_i \bar{\delta}_i^{(1)}$$

$$+ \sum_{t=2}^T \sum_k \sum_l (\mu_{kl} - 1) \left[\psi(\tilde{\mu}_{kl}^{(t)}) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t)} \right) \right]$$

$$+ \sum_{t=2}^T \sum_{i=1}^N \left[\log \bar{\delta}_i^{t-1} - \bar{\delta}_i^{t-1} \bar{\delta}_i^t \right] - E_q \log q. \quad (15)$$

$$\mathscr{L}_t(Z^{(t)}, \delta^{(t)}, \tilde{\mu}^{(t)} | \pi, B, \lambda)$$

$$\approx \sum_{w_{ij}^{(t)}=1} \left(1 + \bar{\delta}_i^{(t)} + \bar{\delta}_j^{(t)} \right) \sum_k \sum_l \phi_{ik}^{(t)} \phi_{jl}^{(t)} \log B_{kl}$$

$$- \sum_{w_{ij}^{(t)}=0} \sum_{k,l} \phi_{ik}^{(t)} \phi_{jl}^{(t)} B_{kl}^{1+\bar{\delta}_i^{(t)}+\bar{\delta}_j^{(t)}} + \sum_i \log \bar{\delta}_i^{(t-1)}$$

$$+ \sum_i \sum_{k,l} \phi_{il}^{(t-1)} \phi_{ik}^{(t)} \left[\psi(\tilde{\mu}_{kl}^{(t)}) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t)} \right) \right]$$

$$+ \sum_i \sum_{k,l} \phi_{ik}^{(t)} \phi_{il}^{(t+1)} \left[\psi\left(\tilde{\mu}_{kl}^{(t+1)} \right) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t+1)} \right) \right]$$

$$+ \sum_{k,l} (\mu_{kl} - 1) \left[\psi\left(\tilde{\mu}_{kl}^{(t)} \right) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t)} \right) \right]$$

$$- \sum_i \bar{\delta}_i^{(t-1)} \bar{\delta}_i^{(t)} - \sum_i \bar{\delta}_i^{(t)} \bar{\delta}_i^{(t+1)} - E_q \log q. \quad (16)$$

In the GEABS model, the latent variables snapshot 1 is depend on that in snapshot 2, variables of snapshot T is

limited by $T-1$, for snapshots $t = 2, \ldots, T-1$, its inference is affected by the combination of the two snapshots before and after it. Here, we only show the general inference process at $t = 2, \ldots, T-1$. The ELBO on one snapshot $t \in [2, T-1]$ is as Eq. (16).

Then, our task is to maximize the $\mathscr{L}_t$ to learn the variational parameters of the latent variables $Z^{(t)}$, $\delta^{(t)}$ and $\tilde{\mu}^{(t)}$. We take the derivatives of $\mathscr{L}_t$ with respect to the variational parameters and set these derivatives to zeros as follows:

$$\nabla \mathscr{L}_t = \left\{ \frac{\partial \mathscr{L}_t}{\partial \phi_{ik}^{(t)}}, \frac{\partial \mathscr{L}_t}{\partial \bar{\delta}_i^{(t)}}, \frac{\partial \mathscr{L}_t}{\partial \tilde{\mu}_{kl}^{(t)}} \right\} = 0, \tag{17}$$

where $\phi_{ik}^{(t)}$ and $\tilde{\mu}_{kl}^{(t)}$ are the approximate posterior distributions for community membership and community transition probability. They meet the constraints in Eq. (18)

$$\sum_{k=1}^{K^{(t)}} \phi_{ik}^{(t)} = 1 \;\; \text{and} \;\; \sum_{l=1}^{K^{(t)}} \tilde{\mu}_{kl}^{(t)} = 1. \tag{18}$$

Then, it is easy to get the update rules as Eqs. (19) and (20)

$$\begin{aligned}
\phi_{ik}^{(t)} \propto \exp \Bigg\{ & \sum_{w_{ij}^{(t)}=1} \left(1 + \bar{\delta}_i^{(t)} + \bar{\delta}_j^{(t)} \right) \sum_l \phi_{jl}^{(t)} \log B_{kl} \\
& - \sum_{w_{ij}^{(t)}=0} \sum_l \phi_{jl}^{(t)} B_{kl}^{1+\bar{\delta}_i^{(t)}+\bar{\delta}_j^{(t)}} \\
& + \sum_k \phi_{ik}^{(t-1)} \left[\psi\left(\tilde{\mu}_{kl}^{(t)} \right) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t)} \right) \right] \\
& + \sum_l \phi_{il}^{(t+1)} \left[\psi\left(\tilde{\mu}_{kl}^{(t+1)} \right) - \psi\left(\sum_l \tilde{\mu}_{kl}^{(t+1)} \right) \right] \Bigg\},
\end{aligned} \tag{19}$$

$$\tilde{\mu}_{kl}^t = \sum_i \phi_{il}^{(t-1)} \phi_{ik}^{(t)} + \mu_{kl}. \tag{20}$$

However, there is a small challenge in deriving $\bar{\delta}_i^{(t)}$ for it has no closed solution. So we take a fast gradient descent method to update it and the gradient is as Eq. (21)

$$\begin{aligned}
\frac{\partial \mathscr{L}_t}{\partial \bar{\delta}_i^{(t)}} = & \sum_{w_{ij}^{(t)}=1} \sum_k \sum_l \phi_{ik}^{(t)} \phi_{jl}^{(t)} \log B_{kl} \\
& - \sum_{w_{ij}^{(t)}=0} \sum_k \sum_l \phi_{ik}^{(t)} \phi_{jl}^{(t)} B_{kl}^{1+\bar{\delta}_i^{(t)}+\bar{\delta}_j^{(t)}} \log B_{kl} \\
& - \bar{\delta}_i^{(t-1)} - \bar{\delta}_i^{(t+1)} - \log \bar{\delta}_i^{(t)} - 1.
\end{aligned} \tag{21}$$

4.2 Variational Inference M-Step

In the E-step, we have updated the variational parameters $Z^{(t)}$, $\delta^{(t)}$, and $\tilde{\mu}^{(t)}$, which could increase the ELBO on the dynamic network. Here, we need to update the model parameters to maximize the log likelihood. Similar to the inference process in the E-step, we are easy to get the update rules for the parameters

$$\pi_k \propto \sum_i \phi_{ik}^{(1)}, \;\; \lambda = \frac{1}{N} \sum_i \bar{\delta}_i^{(1)}, \;\; B_{kl} = \propto +\alpha \frac{\partial \mathscr{L}(B_{kl})}{\partial B_{kl}}, \tag{22}$$

where α is the learning rate and the gradient for B_{kl} is calculated by

$$\begin{aligned}
\frac{\partial \mathscr{L}(B_{kl})}{\partial B_{kl}} = & \frac{\sum_{t=1}^{T} \sum_{w_{ij}^{(t)}=1} (1 + \bar{\delta}_i^{(t)} + \bar{\delta}_j^{(t)}) \phi_{ik}^{(t)} \phi_{jl}^{(t)}}{b_{kl}} \\
& - \sum_{t=1}^{T} \sum_{w_{ij}^{(t)}=0} (1 + \bar{\delta}_i^{(t)} + \bar{\delta}_j^{(t)}) \phi_{ik}^{(t)} \phi_{jl}^{(t)} B_{kl}^{\bar{\delta}_i^{(t)}+\bar{\delta}_j^{(t)}}.
\end{aligned} \tag{23}$$

Considering the computational efficiency in gradient descent, here we simplify the calculation of association probability matrix of community B with Eq. (24)

$$B_{kl} = \frac{\sum_{t=1}^{T} \sum_{i \sim j} (\phi_{ik}^{(t)} \phi_{jl}^{(t)} + \phi_{il}^{(t)} \phi_{jk}^{(t)}) W_{ij}^{(t)}}{\sum_{t=1}^{T} \sum_{i \sim j} (\phi_{ik}^{(t)} \phi_{jl}^{(t)} + \phi_{il}^{(t)} \phi_{jk}^{(t)})}. \tag{24}$$

In fact, we ignore the influence of node popularity on the B to speed up computing. In addition, if the number of communities are varying of snapshots, we could take $B^{(t)} \in (0, 1)^{K^{(t)} \times K^{(t)}}$ for each snapshot instead of a common B. With variational and model parameters are updated iteratively. After convergence, we can get the last results, besides, the community label $z^{(t)}$ and transition probability $A^{(t)}$ are calculated by Eq. (25)

$$z_i^{(t)} = \arg \max_k \phi_{ik}^{(t)}, \quad A_{kl}^{(t)} \propto \arg \tilde{\mu}_{kl}^{(t)}, \tag{25}$$

besides, $q(\delta_i^t)$ is a degenerated distribution with the parameter $\bar{\delta}_i^t$, so the parameter $\delta_i^t = \bar{\delta}_i^t$.

4.3 The Algorithm

With the inference process, we propose the procedure for maximizing the ELBO $\mathscr{L}(q)$ and the algorithm is summarized as shown in Algorithm 3.

Algorithm 3. Optimization Process of $\mathcal{L}$

Input: The similarity matrix $W^{(t)}$, the number of communities $K^{(t)}$ for each snapshot and the stop criterion ε.
Output: $\bar{\delta}^{(t)}, \pi, B, A^{(t)}, \lambda$ and $Z^{(t)}$
1: Initialization model parameters π, B, λ, μ
2: Sample variational parameters $\phi^{(t)}, \bar{\delta}^{(t)}$ and $\tilde{\mu}$
3: **repeat**
4:    **for** each snapshot t **do**
5:      **variational E-step**
6:      Update $\phi_{ik}^{(t)}$ via Eq. (19)
7:      Update $\tilde{\mu}$ via Eq. (20)
8:      Update $\bar{\delta}_i^{(t)}$ by coordinate gradient ascend and gradient via Eq. (21)
9:      **variational M-step**
10:      Update π via Eq. (22)
11:      Update B via Eq. (24)
12:      Compute variational likelihood $\mathcal{L}^{new}$ with updated parameters by Eq. (15)
13:    **end for**
14: **until** $|\mathscr{L}^{new} - \mathscr{L}^{old}| < \varepsilon$
15: **for** every time t **do**
16:    Get the community label $z_i^{(t)}$ of each node and calculate the community transition parameter A_{kl}^t in each community via Eq. (25) and return $\delta_i^{(t)}$.
17: **end for**

4.4 Complexity Analysis

The complexity of GEABS depends on the steps of updating ϕ, i.e., $O(TK^2N^2)$, where T is the number of snapshots, n is the number of nodes in the network, and K is the average number of communities. In theory, the complexity is not greater than DSBM, which is $O(N^2)$. Since most real-world networks are sparse, we can also improve the efficiency of GEABS by using negative sampling and parallelism in operations.

5 EXPERIMENT AND ANALYSIS

In this section, we verify the proposed method on both the and synthetic and real-world dynamic networks, including detecting abnormal behaviors caused by changes in the macroscopic, mesoscopic, and micro-scale levels of the network. We compare our results with some baselines on the different tasks (community detection, anomaly detection on network level, community level, and node level). Besides, we also show a case study to verify the effectiveness of the proposed model.

5.1 Datasets

The experimental part involves one synthetic network and three real-world networks, namely, Kit-email network, Enron email network, and World trade network. Their details are as follows:

- *Synthetic Data [45]*: For this dataset, several community-level events are introduced to make it more similar to real-world networks. We use this dataset to evaluate the effectiveness of the community-level anomaly parameter A. To achieve this goal, we generate a dynamic network with merge and split events in consecutive snapshots. This network contains 10 snapshots, 250 nodes, and a fixed community number of 22.
- *Kit-email network [46]*: This email network is composed of 1,097 email IDs (the number of nodes) and 27,887 messages. We use it to construct a temporal network with time intervals of 6 months, and the number of snapshots of this temporal network is 8. The number of communities is 27. Nodes in this network represent senders and recipients and the edges denote the relationships.
- *Enron email network [47]*: The Enron network is constructed by email communications between 151 senior managers of Enron Energy, which filed for bankruptcy in 2001 after it was found to be fraudulent. We extracted a dynamic social network with 12 time snapshots by month, which containing 151 nodes. Each time snapshot includes monthly communications of senior management, the edges linking the senders and receivers of emails. In addition, according to the company's development timeline, we identified 7 incidents, which largely affected the structure of the Enron mail network.
- *World trade network [48]*: World trade network data records the total annual import and export trade volume of 196 countries from the year 1948 to 2000. We

sorted the data into a dynamic network, which included 196 nodes, 5,735 edges, and 53 time slices. The edge of each time slice is the total annual trade volume between the two countries.

5.2 Community Detection

For community detection, we compare our method GEABS with 4 popular baseline methods, which contains:

- *ECD [49]*: It combines the proposed new genetic operator and classic genetic operators to exploit inter and intra connections between nodes. This approach improves the discovery of evolving community structures and finds the best balance between clustering accuracy and temporal smoothness.
- *DECS [50]*: It is a novel algorithm based on genome representation, employing Population Generation via Label Propagation (PGLP) for population initialization and decomposition framework for multi-objective optimization.
- *PisCES [51]*: From the perspective of spectral optimization, this is a global method that can infer the evolution by combining a series of networks, eigenvector smoothing, and degree correction.
- *DSBM [52]*: It is the most successful generative model for dynamic community detection and evolution analysis based on SBM.

We evaluate the community detection performance of our model and baselines within three performance metrics in both synthetic dataset and real-world dataset kit-email.

5.2.1 Evaluation Metrics on Community Detection

Accuracy (AC) or error rate [53] is usually denoted as the distance between the ground truth and community membership of one method, which is defined as

$$AC = \|ZZ^T - Z'Z'^T\|_F, \qquad (26)$$

where Z and Z' are the community membership of ground truth, respectively. $\|\cdot\|_F$ is the Frobenius norm, the smaller the AC value on each snapshot, the better the community results.

As most temporal community detection work [54] does, we also use Normalized Mutual Information (NMI) as one of our performance metrics to evaluate the proposed model and baselines for community detection in dynamic networks. Because NMI is specifically designed for static networks, we calculate it for different methods at each snapshot of the dynamic network. NMI is used when there exists ground truth, which measures the similarity between a given community partition and the true community structure. Let $Z = \{Z_1, \ldots, Z_K\}$ and $Z' = \{Z'_1, \ldots, Z'_K\}$ represent the true community partition and the community partition to be evaluated, respectively, where Z_k or Z'_k is the node set of community k. For Z and Z', we usually have $Z_k \cap Z_l = \emptyset, k \neq l$ and $\bigcup Z_k$ is the node set of dynamic networks. The NMI is denoted as
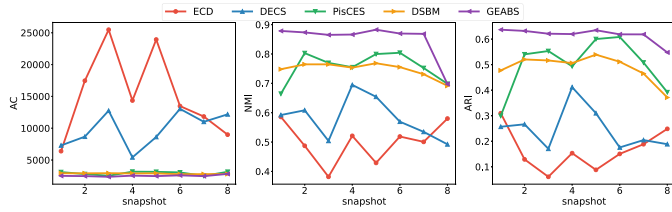
Fig. 3. Community detection result in kit-email data.

Fig. 4. Community detection result in synthetic data.

$$\text{NMI}(Z, Z') = \frac{\sum_{Z,Z'} p(Z, Z') \log \frac{p(Z,Z')}{p(Z)p(Z')}}{\max(H(Z), H(Z'))}, \tag{27}$$

where $H(Z)$ and $H(Z')$ are the entropy of community Z and Z', respectively. The value of NMI is between 0 and 1. A higher value of NMI indicates better community detection performance.

Adjusted Rand index is another metric for clustering and community detection performance, which is defined as

$$ARI = \frac{RI - \mathbb{E}[RI]}{\max(RI) - \mathbb{E}[RI]}, \tag{28}$$

where $RI = \frac{a+b}{\binom{n}{2}}$ is the Rand Index (RI) value of a community, a and b are the number of pairs of nodes placed in the same cluster and in different clusters, respectively. $\mathbb{E}$ is the expectation operator. A larger ARI value indicates better performance on community detection.

5.2.2 Community Detection Result

The community detection results of our model and baselines in the Synthetic dataset and kit-email dataset are shown in Figs. 3 and 4, respectively. Our method achieves the best performance in both synthetic and real-world
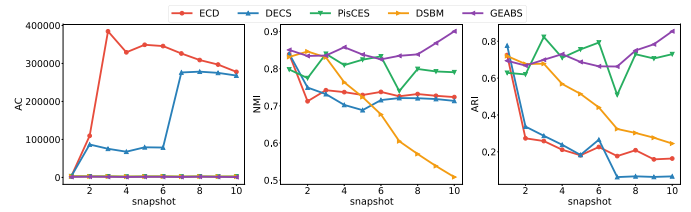
datasets due to the well-defined generative mechanism. It is worth mentioning that our method achieves better AC, NMI and ARI results in the real-world network compared to the synthetic network. Although the synthetic network is generated by a mechanism of imitating the real world, its network structure is still very regular compared to the real-world network.

5.3 Network-Level Anomaly

We use our method to detect change points of the two empirical temporal networks, the Enron email network, and the World Trade Network. The structure of the network is very likely to be affected by some known external events (Table 2), which are used as the truth of the change points, i.e., the network level anomaly point. To verify our results, we choose six approaches that represent the most advanced in change point detection of dynamical networks as baseline methods, including:

- *SCOUT [18]*: This method realizes synchronous detection of change points and communities. It uses the existing search strategy to find the change point in the time series network, performs consensus clustering on the network in the same segment, and

TABLE 2
The Social Events in Related Datasets

Datasets	Date	Event (change point)
Enron	2001-02	Tom White resigns from EES[1]
	2001-04	Quarterly Conference Call
	2001-06	FERC finally institutes price caps across the western states. The California energy crisis ends.
	2001-07	Skilling announces desire to resign.
	2001-08	Skilling resigns as CEO
	2001-09	Skilling sells stock
	2001-11	Stock prices plummet
World Trade	1950	Third Gatt round held in Torquay, England, where countries exchanged some 8,700 tariff concessions, cutting the 1948 tariff levels by 25%
	1955	The next trade round completed in May 1956, resulting in $2.5bn in tariff reductions.
	1960	European Free Trade Association (FEAT) established
	1964	The Kennedy Round, named in honor of the late US president, achieves tariff cuts worth $40 billion of World Trade.
	1967	Founding of the European Community
	1968	Two important ISO (International Organization for Standardization) recommendations standardized containerization globally [55]
	1971	The Ranger Committee is formed
	1973	Oil shock
	1983	The resumption of trade across the Sino–Central Asian border in 1983 accompanied a gradual thaw in relations between China and the Soviet Union [56]
	1991	Collapse of the Soviet Union [57]
	1999	The birth of the euro affected the trade balance of the eurozone

finally uses the constraint function to find the change point and community.

- *CICPD [41]*: It is a novel change point detection method based on community detection. It uses the PageRank algorithm to learn the network representation of each time slice, and calculates the distance between them, and generates a new network. Perform spectral clustering on the new network to detect change points.
- *GHRG [40]*: It is the first to use the online probabilistic learning framework to solve the problem of change points, and use a hierarchical random graph to generate a network with a Bayesian hypothesis test to quantify the occurrence of change points.
- *LAD [43]*: It detects the change points and events of the dynamic network at the same time, which is a spectral-based method. The core idea of LAD is to summarize entire graph snapshots into low dimensional embeddings through the singular values of the graph Laplacian, and to explicitly model the short-term and long-term behavior of dynamic network evolution.
- *NetWalk [39]*: This method uses learning network representation to detect dynamic network change points, which can be dynamically updated with the development of the network. It encodes the vertices of the dynamic network into a vector representation through group embedding, which together minimizes the pairwise distance of each wandering vertex derived from the dynamic network, and realizes the detection of the dynamic network change point through clustering-based technology.

5.3.1 Evaluation Metrics on Change Point

Change point detection can be used as a binary classification problem, so we can calculate the scores of *Precision*, *Recall* and F_1 to evaluate the performance of our model, which are separately defined as

$$Precision = \frac{TP}{TP + FP}, \tag{29}$$

where *Precision* quantifies the number of positive class predictions that actually belong to the positive class

$$Recall = \frac{TP}{TP + FN}, \tag{30}$$

where *Recall* quantifies the number of positive class predictions made out of all positive examples in the dataset

$$F_1 = \frac{2TP}{2TP + FP + FN}, \tag{31}$$

where the comprehensive measurement F_1 provides a single score that balances both the concerns of precision and recall in one number.

These values are in the range of 0 to 1. A score of 1 indicates a perfection spotting of change points, while a score of 0 implies that no change points being detected at the other

TABLE 3
Precision, Recall and F1 of Our Method and the Baseline Methods

Method	Enron			World Trade		
	Precision	Recall	F_1	Precision	Recall	F_1
SCOUT	**1.0000**	0.1818	0.3077	0.4000	0.1818	0.2500
CICPD	0.5714	0.5714	0.5714	0.3333	0.4545	0.3846
GHRG	0.4284	0.1667	0.2400	0.3500	0.6364	0.4516
LAD	0.6667	0.5714	0.6154	0.2222	0.3636	0.2758
NetWalk	0.8333	**0.7143**	0.7692	0.7333	0.8181	0.7734
GEABS	**1.0000**	**0.7143**	**0.8333**	**0.7143**	**0.9091**	**0.8000**

extreme. The change point in our model is calculated by extracting the local maximum value of $E_{nl}^{(t)}$.

5.3.2 Network-Level Anomaly Results

As shown in Table 3, our method GEABS has the best performance among all baseline methods. The Precision of GEABS in Enron data is 1, which indicates that all the anomaly points we find are true. GEABS has about 0.91 recall rate in World Trade, which means that our method finds almost all the anomaly points in this data. It is worth mentioning that, just as shown in Fig. 6, we calculate the hit rate of the top k network-level anomaly value to the anomaly events in the Enron dataset and world trade dataset of GEABS, NetWalk and LAD. Due to the complexity of World trade dataset, NetWalk and LAD do not perform well. But in Enron dataset, NetWalk has a good hit rate. But the top 10 hit rates of both NetWalk and LAD are not better than GEABS. As for GEABS, the hit rate of top 5 network anomaly values in the Enron dataset is 100%, while in the world trade dataset, the hit rate of top 3 network anomaly values is 100%, which proves the effectiveness of our network-level anomaly detection scheme and the correctness of our method.

5.4 Community-Level Anomaly

We use the synthetic network to measure the ability to capture the transition tendency of communities of our model parameter A. We compare the transition tendency matrix A of our model and the true transition behavior in this dataset, and the transition behavior of baseline method PisCES, which has the best community detection performance. The result is shown in Fig. 5. Our method can capture the community-level transition tendency of a network, while PisCES cannot. So our model can handle community events such as community merge, split, expand and shrink and so on. This is because the transition matrix A can accurately capture the future behavior of nodes within the community.

Furthermore, as mentioned above, GEABS can handle different community numbers in dynamic networks. More specifically, the ground truth of community membership is unknown in a real-world network. Communities may merge, split and even die, and all community events may lead to changes in the number of communities. So we test our method in Enron dataset with different community numbers at different snapshots. The community number
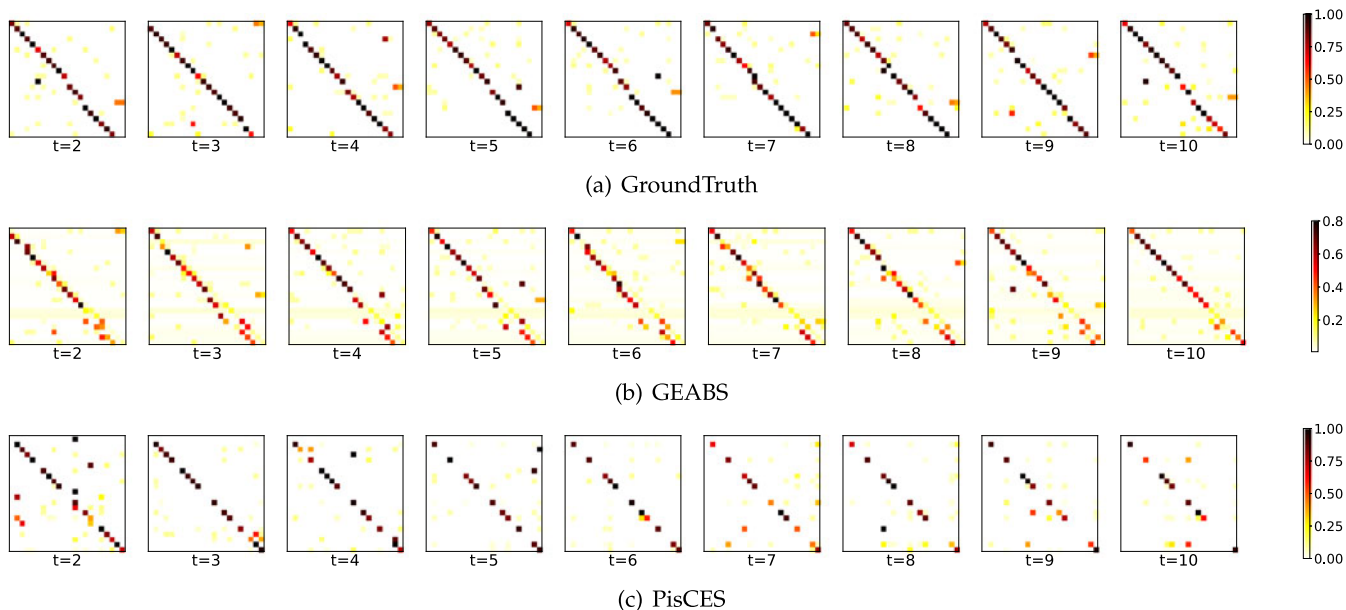
(a) GroundTruth

(b) GEABS

(c) PisCES

Fig. 5. Visualization of community transition matrix in synthetic Data. Each row in the graph represents the transfer tendency of the node in the corresponding community in the current snapshot. The darker the color, the more transfer tendency is.

is initialized by the static community detection method. As shown in Fig. 9, the community membership transition of nodes in Enron data is violent, and the community number changes at $t = 5, 6, 7, 8, 9$. As shown in Table 2, compared to the event of Enron data, we can find that there are change points at $t = 6, 7, 8, 9$, indicating that the design of GEABS can handle the ever-changing number of communities, which is essential for network anomaly detection.

5.5 Node Anomaly Analysis

It is hard to judge the node anomaly behavior due to the lack of reasonable evaluation metrics and datasets. To evaluate the effectiveness of the node anomaly index, $E_{cl(i)}$, we defined based on our model parameter δ, we draw a scatter chart of node anomaly index in international trade network to heuristically verify the validity of our parameters $E_{cl(i)}$ through world-historical information. Just as shown in Fig. 7, after World War II, America (USA) is the most stable country in the world and the leader of Western countries. Therefore, the abnormal value of its node reaches the lowest. Japan (JPN) also has a lower anomaly value than most of the other countries, this is also a case in point for our node anomaly

index $E_{cl(i)}$. Because up to 2,000, Japan has the second most GDP in the world.

In addition, small countries like Haiti, Saint Lucia, and Dominica always have the highest node anomaly value. This is because these countries do not have competitive goods, which makes their trade always fluctuate.

5.6 Case Study

To further prove the anomaly detection ability of our model, we visualize the dynamic evolution detected from our model of the international trade dataset in macro-, micro- and mesoscopic simultaneously. As shown in Fig. 8, different network-level anomaly indexes may be caused by different factors. Such as in 1950, the third Gatt round held in England, which makes several International trade group combines in that year. Besides, at the node level, we notice that China (CHN) has a very high δ value. By our investigation, the People's Republic of China was founded in 1949, therefore, the most frequent period of international trade growth of China is 1950.
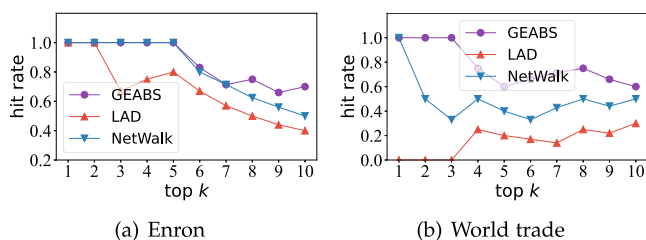

(a) Enron　　　　(b) World trade

Fig. 6. The hit rate of the top k network-level anomaly value to the anomaly events of GEABS compared to LAD and NetWalk.
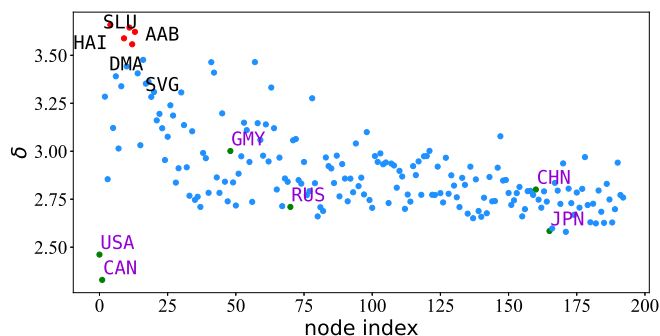

Fig. 7. Node anomaly parameters in international trade data. The red dots represent the countries with the largest 5 node anomaly values, while the green dots represent the major countries we select (America, Canada, Germany, Russia, China and Japan).
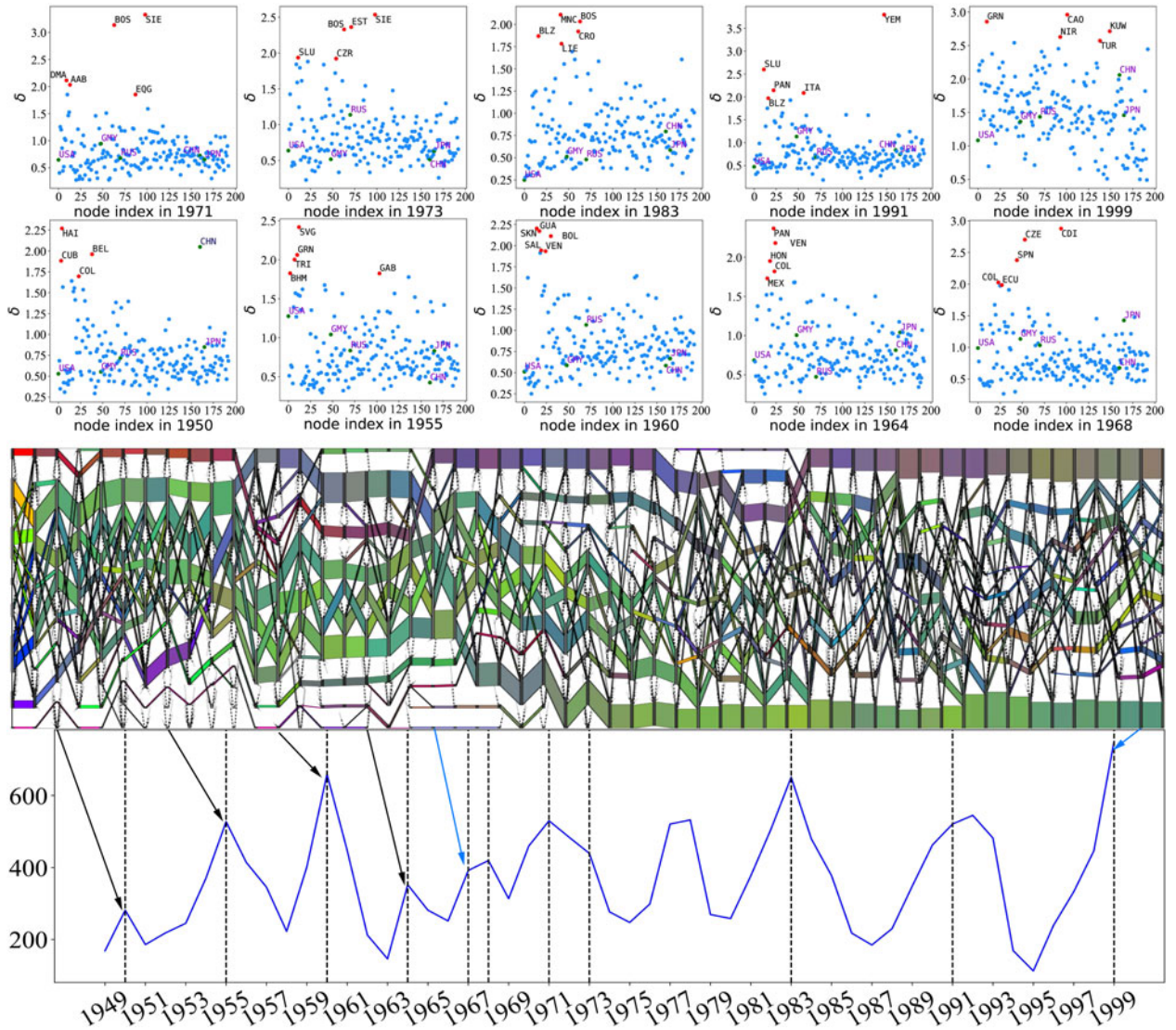
Fig. 8. A case study of anomaly detection in world trade data. The red dots in the node-level scatter charts represent the countries with the largest 5 node anomaly values, while the green dots represent the major countries we select (America, Germany, Russia, China, and Japan). And the community-level Sankey chart is a community evolution visualization based on the transition matrix A and the community membership Z. Finally, The network-level line chart is the visualization of network-level anomaly value $E_{nl}^{(t)}$, the local peak in this plot is the anomaly point detected from our model, which is a change point in the dynamic network.
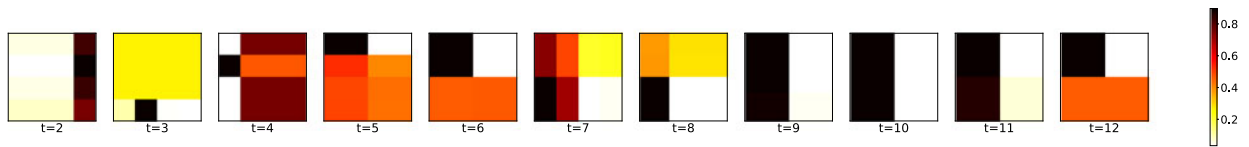

Fig. 9. Transition matrix of GEABS on Enron data with changing number of communities K at different snapshot t.

The oil shock in 1973 makes the δ value of Russia rise sharply. Since the main export in Russia is oil, the oil shock has a great influence on Russia, while it has little impact on other main countries since these countries have multiple types of trade imports and exports. Furthermore, the Disintegration of the Soviet Union even did not affect Russia's exports and imports, but this event makes several country groups split.

In addition, we find that every important event always makes small countries have the highest δ values. This indicates that small countries are always more difficult to survive than large countries during worldwide events. Moreover, anomalies at the community level and the network level are closely connected. Most of the change points in Fig. 8 contain major changes in the community, such as $1949, 1955, 1983$. However, node-level anomalies are always affected by community or network events, such as the small countries we mentioned above. This phenomenon can reveal the relationship between different levels of networks and is worthy of further study.

6 CONCLUSION

Anomalies in dynamic networks are very complex. We divided these anomalies into three levels: macro level, mesoscopic level, and micro level, as well as formalized the description of the specific tasks in each level. This paper proposed an anomaly detection framework GEABS in dynamic networks, which can detect anomalies at different levels and realize the discovery of structural changes in the evolution of communities. By assuming that the network follows an SBM, node popularity and community transition matrix are generated to measure node and community anomalies through a defined operator. We can also detect the network evolution anomaly by combining the node level popularity and the community level transition parameter A and community structure Z. The experimental results show that our model has indeed achieved superior results compared with existing methods.

The anomalies in dynamic networks could benefit many pieces of research in the future [58], e.g., Blockchain [59], which mainly deals with a large amount and variety of data, can be used to capture anomalies in dynamic networks and provide early warnings for unsafe data. In addition, this research can also help to better apply 5G technology [60] by detecting abnormal devices in high-density communication devices. Thus, our research also could be combined with various studies related to data security. Since our model is based on a probabilistic graphical model and variational inference, some improvements can be made in terms of complexity. Furthermore, in our future work, we consider improving the anomaly detection performance in dynamic networks from the following aspects: 1) Use graph neural networks to build generative models to capture different levels of anomalies; 2) Consider using incremental or stream computing to speed up the calculations on the network; 3) Consider fusing text information or performing anomaly detection in heterogeneous networks to achieve more complex dynamic network anomaly detection.

REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009.

[2] H. Zenati, M. Romain, C.-S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *Proc. IEEE Int. Conf. Data Mining*, 2018, pp. 727–736.

[3] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Gener. Comput. Syst.*, vol. 55, pp. 278–288, Feb. 2016.

[4] S. Amraee, A. Vafaei, K. Jamshidi, and P. Adibi, "Anomaly detection and localization in crowded scenes using connected component analysis," *Multimedia Tools Appl.*, vol. 77, no. 12, pp. 14 767–14 782, 2018.

[5] H. Kasai, W. Kellerer, and M. Kleinsteuber, "Network volume anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 3, pp. 636–650, Sep. 2016.

[6] Q. Zhou, L. Li, and H. Tong, "Towards real time team optimization," in *Proc. IEEE Int. Conf. Big Data*, 2019, pp. 1008–1017.

[7] Y. Lv, Y. Chen, X. Zhang, Y. Duan, and N. L. Li, "Social media based transportation research: The state of the work and the networking," *IEEE/CAA J. Automatica Sinica*, vol. 4, no. 1, pp. 19–26, Jan. 2017.

[8] M. Ley, "The DBLP computer science bibliography: Evolution, research issues, perspectives," in *Proc. Int. Symp. String Process. Inf. Retrieval*, 2002, pp. 1–10.

[9] K. Ding, J. Li, and H. Liu, "Interactive anomaly detection on attributed networks," in *Proc. 12th ACM Int. Conf. Web Search Data Mining*, 2019, pp. 357–365.

[10] K. Ding, J. Li, N. Agarwal, and H. Liu, "Inductive anomaly detection on attributed networks," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, 2020, pp. 1288–1294.

[11] S. Ranshous, S. Shen, D. Koutra, S. Harenberg, C. Faloutsos, and N. F. Samatova, "Anomaly detection in dynamic networks: A survey," *Wiley Interdisciplinary Rev., Comput. Statist.*, vol. 7, no. 3, pp. 223–247, 2015.

[12] H. Huang, J. Tang, L. Liu, J. Luo, and X. Fu, "Triadic closure pattern analysis and prediction in social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 12, pp. 3374–3389, Dec. 2015.

[13] R. Hassanzadeh and R. Nayak, "A semi-supervised graph-based algorithm for detecting outliers in online-social-networks," in *Proc. 28th Annu. ACM Symp. Appl. Comput.*, 2013, pp. 577–582.

[14] N. A. Heard, D. J. Weston, K. Platanioti, and D. J. Hand, "Bayesian anomaly detection methods for social networks," *Ann. Appl. Statist.*, vol. 4, pp. 645–662, 2010.

[15] L. Xiao, Y. Zhang, Z. Hu, and J. Dai, "Performance benefits of robust nonlinear zeroing neural network for finding accurate solution of Lyapunov equation in presence of various noises," *IEEE Trans. Ind. Informat.*, vol. 15, no. 9, pp. 5161–5171, Sep. 2019.

[16] L. Xiao, B. Liao, S. Li, Z. Zhang, L. Ding, and L. Jin, "Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 98–105, Jan. 2018.

[17] L. Xiao et al., "A new noise-tolerant and predefined-time ZNN model for time-dependent matrix inversion," *Neural Netw.*, vol. 117, pp. 124–134, 2019.

[18] Y. Hulovatyy and T. Milenković, "SCOUT: Simultaneous time segmentation and community detection in dynamic networks," Sci. Rep., vol. 6, 2016, Art. no. 37557.

[19] R. C. Y. Cheung, A. Aue, S. Hwang, and T. C. M. Lee, "Simultaneous detection of multiple change points and community structures in time series of networks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 580–591, Aug. 2020.

[20] H. Wang and C. Qiao, "A nodes' evolution diversity inspired method to detect anomalies in dynamic social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 10, pp. 1868–1880, Oct. 2020.

[21] H. Cheng, P.-N. Tan, C. Potter, and S. Klooster, "A robust graph-based algorithm for detection and characterization of anomalies in noisy multivariate time series," in *Proc. IEEE Int. Conf. Data Mining Workshops*, 2008, pp. 349–358.

[22] R. K. Darst, C. Granell, A. Arenas, S. Gómez, J. Saramäki, and S. Fortunato, "Detection of timescales in evolving complex systems," Sci. Rep., vol. 6, no. 1, pp. 1–8, 2016.

[23] L. Zheng, Z. Li, J. Li, Z. Li, and J. Gao, "AddGraph: Anomaly detection in dynamic graph using attention-based temporal GCN," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 4419–4425.

[24] C. C. Aggarwal, Y. Zhao, and P. S. Yu, "Outlier detection in graph streams," in *Proc. IEEE 27th Int. Conf. Data Eng.*, 2011, pp. 399–409.

[25] S. Ranshous, S. Harenberg, K. Sharma, and N. F. Samatova, "A scalable approach for outlier detection in edge streams using sketch-based approximations," in *Proc. SIAM Int. Conf. Data Mining*, 2016, pp. 189–197.

[26] J. Sun, Y. Xie, H. Zhang, and C. Faloutsos, "Less is more: Compact matrix decomposition for large sparse graphs," in *Proc. SIAM Int. Conf. Data Mining*, 2007, pp. 366–377.

[27] K. Shin, B. Hooi, J. Kim, and C. Faloutsos, "D-Cube: Dense-block detection in terabyte-scale tensors," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, 2017, pp. 681–689.

[28] K. Shin, B. Hooi, and C. Faloutsos, "M-Zoom: Fast dense-block detection in tensors with quality guarantees," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2016, pp. 264–280.

[29] M. Mongiovi, P. Bogdanov, and A. K. Singh, "Mining evolving network processes," in *Proc. IEEE 13th Int. Conf. Data Mining*, 2013, pp. 537–546.

[30] B. A. Miller, N. Arcolano, and N. T. Bliss, "Efficient anomaly detection in dynamic, attributed graphs: Emerging phenomena and Big Data," in *Proc. IEEE Int. Conf. Intell. Secur. Informat.*, 2013, pp. 179–184.

[31] D. Eswaran, C. Faloutsos, S. Guha, and N. Mishra, "SpotLight: Detecting anomalies in streaming graphs," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1378–1386.

[32] N. Du, X. Jia, J. Gao, V. Gopalakrishnan, and A. Zhang, "Tracking temporal community strength in dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 11, pp. 3125–3137, Nov. 2015.

[33] A. Karaaslanlı and S. Aviyente, "Community detection in dynamic networks: Equivalence between stochastic blockmodels and evolutionary spectral clustering," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 130–143, Feb. 2021.

[34] X. Han, Y. Zhao, and M. Small, "Identification of dynamical behavior of pseudoperiodic time series by network community structure," *IEEE Trans. Circuits Syst., II, Exp. Briefs*, vol. 66, no. 11, pp. 1905–1909, Nov. 2019.

[35] S. Y. Bhat and M. Abulaish, "HOCTracker: Tracking the evolution of hierarchical and overlapping communities in dynamic social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 4, pp. 1019–1013, Apr. 2015.

[36] P. Bródka, S. Saganowski, and P. Kazienko, "GED: The method for group evolution discovery in social networks," *Soc. Netw. Anal. Mining*, vol. 3, no. 1, pp. 1–14, 2013.

[37] X. Ma and D. Dong, "Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 5, pp. 1045–1058, May 2017.

[38] B. Baingana and G. B. Giannakis, "Joint community and anomaly tracking in dynamic networks," *IEEE Trans. Signal Process.*, vol. 64, no. 8, pp. 2013–2025, Apr. 2016.

[39] W. Yu, W. Cheng, C. C. Aggarwal, K. Zhang, H. Chen, and W. Wang, "NetWalk: A flexible deep embedding approach for anomaly detection in dynamic networks," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 2672–2681.

[40] L. Peel and A. Clauset, "Detecting change points in the large-scale structure of evolving networks," in *Proc. AAAI Conf. Artif. Intell.*, 2015, pp. 2914–2920 .

[41] T. Zhu, P. Li, L. Yu, K. Chen, and Y. Chen, "Change point detection in dynamic networks based on community identification," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 3, pp. 2067–2077, Third Quarter 2020.

[42] D. Koutra, J. T. Vogelstein, and C. Faloutsos, "DELTACON: A principled massive-graph similarity function," in *Proc. SIAM Int. Conf. Data Mining*, 2013, pp. 162–170.

[43] S. Huang, Y. Hitti, G. Rabusseau, and R. Rabbany, "Laplacian change point detection for dynamic graphs," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 349–358.

[44] G. Schwarz et al., "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, 1978.

[45] D. Greene, D. Doyle, and P. Cunningham, "Tracking the evolution of communities in dynamic social networks," in *Proc. Int. Conf. Advances Soc. Netw. Anal. Mining*, 2010, pp. 176–183.

[46] R. Görke, M. Holzer, O. Hopp, J. Theuerkorn, and K. Scheibenberger, "Dynamic network of email communication at the department of informatics at Karlsruhe institute of technology," 2011. Accessed: Dec. 19, 2021. [Online]. Available: i11www.iti.kit.edu/ projects/spp1307/emaildata

[47] G. J. Benston and A. L. Hartgraves, "Enron: What happened and what we can learn from it," *J. Accounting Public Policy*, vol. 21, no. 2, pp. 105–127, 2002.

[48] K. S. Gleditsch, "Expanded trade and GDP data," 2002. Accessed: Dec. 19, 2021. [Online]. Available: http://ksgleditsch.com/ exptradegdp.html

[49] F. Liu, J. Wu, C. Zhou, and J. Yang, "Evolutionary community detection in dynamic social networks," in *Proc. Int. Joint Conf. Neural Netw.*, 2019, pp. 1–7.

[50] F. Liu, J. Wu, S. Xue, C. Zhou, J. Yang, and Q. Sheng, "Detecting the evolving community structure in dynamic social networks," *World Wide Web*, vol. 23, no. 2, pp. 715–733, 2020.

[51] F. Liu, D. Choi, L. Xie, and K. Roeder, "Global spectral clustering in dynamic networks," *Proc. Nat. Acad. Sci. USA*, vol. 115, no. 5, pp. 927–932, 2018.

[52] T. Yang, Y. Chi, S. Zhu, Y. Gong, and R. Jin, "Detecting communities and their evolutions in dynamic social networks—A Bayesian approach," *Mach. Learn.*, vol. 82, no. 2, pp. 157–189, 2011.

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

[54] T. Hartmann, A. Kappes, and D. Wagner, "Clustering evolving networks," in *Algorithm Engineering*. Berlin, Germany: Springer, 2016, pp. 280–329.

[55] A. Rushton, P. Croucher, and P. Baker, *The Handbook of Logistics and Distribution Management: Understanding the Supply Chain*. London, U.K.: Kogan Page Publishers, 2014.

[56] H. H. Karrar, "The resumption of Sino–central asian trade, c. 1983–94: Confidence building and reform along a cold war fault line," *Central Asian Surv.*, vol. 35, no. 3, pp. 334–350, 2016.

[57] J. C. Cuaresma and M. Roser, "Borders redrawn: Measuring the statistical creation of international trade," *World Economy*, vol. 35, no. 7, pp. 946–952, 2012.

[58] S. Tuli et al., "HealthFog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments," *Future Gener. Comput. Syst.*, vol. 104, pp. 187–200, 2020.

[59] S. S. Gill et al., "Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," *Internet Things*, vol. 8, 2019, Art. no. 100118.

[60] R. I. Ansari et al., "A new dimension to spectrum management in IoT empowered 5G networks," *IEEE Netw.*, vol. 33, no. 4, pp. 186–193, Jul./Aug. 2019.
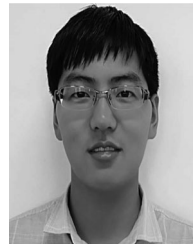
Pengfei Jiao received the PhD degrees in computer science from Tianjin University, Tianjin, China, in 2018. From 2018 to 2021, he was a lecture with the Center of Biosafety Research and Strategy of Tianjin University. He is currently a professor with the School of Cyberspace, Hangzhou Dianzi University, Hangzhou, China. His current research interests include complex network analysis and its applications.

Tianpeng Li received the BE and MS degrees from Tianjin University, Tianjin, China, in 2017 and 2020, respectively. He is currently working toward the PhD degree with the College of Intelligence and Computing, Tianjin University, Tianjin, China. His research interests include complex network analysis, dynamic community detection, and network embedding.

Yingjie Xie received the MS degree from the College of Intelligence and Computing, Tianjin University, Tianjin, China, in 2019. She is currently working toward the PhD degree with the School of College of Intelligence and Computing, Tianjin University, Tianjin, China. Her research interests include machine learning, dynamic community detection, community evolution in dynamic network, detection change points and its applications in computer science.

Yinghui Wang received the master's degree from Tianjin University, Tianjin, China, in 2018. She is currently working toward the EngD degree with the School of Computer Science and Technology, Tianjin University, Tianjin, China. Her research interests include complex network analysis and computational social science.

Wenjun Wang is currently a professor with the School of College of Intelligence and Computing, Tianjin University, chief expert of major projects of the National Social Science Foundation, the Big Data specially-invited expert of Tianjin Public Security Bureau and the director of the Tianjin Engineering Research Center of Big Data on Public Security. His research interests include computational social science, large-scale data mining, intelligence analysis, and multi-layer complex network modeling. He was the principal investigator or was responsible for more than 50 research projects, including the Major Project of National Social Science Fund, the Major Research Plan of the National Natural Science Foundation, the National Science–Technology Support Plan Project of China, etc. He has published more than 50 papers on main international journals and conferences.

Dongxiao He received the PhD degrees in computer science from Jilin University, Changchun, China, in 2014. She is currently an associate professor with the College of Intelligence and Computing, Tianjin University, Tianjin, China. She has published more than 50 international journal and conference papers. Her current research interests include data mining and analysis of complex networks.

Huaming Wu received the BE and MS degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2009 and 2011, respectively, and the PhD (highest honor) degree in computer science from Freie Universität Berlin, Berlin, Germany, in 2015. He is currently an associate professor with the Center for Applied Mathematics, Tianjin University, China. His research interests include wireless networks, mobile edge computing, Internet of Things, and complex networks.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.