Community detection via measuring the strength between nodes for dynamic networks

Kai Yang, Qiang Guo, Jian-Guo Liu

PII: S0378-4371(18)30762-3
DOI: https://doi.org/10.1016/j.physa.2018.06.038
Reference: PHYSA 19732

To appear in: *Physica A*

Received date : 4 February 2018
Revised date : 13 April 2018

Highlights (for review)

- We present community detection method based on nonnegative matrix factorization for dynamic networks considering the strength between nodes.
- The node pairs with stronger connection strength are set have more possibility to be grouped into the same community.
- The accuracy of our algorithm improve 0.3425, 0.5191 for the synthetic networks.

Community detection via measuring the strength between nodes for dynamic networks

Kai Yang [a], Qiang Guo [a], Jian-Guo Liu* [b]

[a]*Research Center of Complex Systems Science, University of Shanghai for Science and Technology, Shanghai 200093, PR China*

[b]*Data Science and Cloud Service Research Centre, Shanghai University of Finance and Economics, Shanghai 200433, PR China*

Abstract

The detection of community structure for dynamic social networks is significant for understanding evolution features of collective behaviors. In this paper, we present community detection method based on nonnegative matrix factorization for dynamic networks considering the strength between nodes. The basic idea of this algorithm is that node pairs with stronger connection strength have more possibility to be grouped into the same community. Firstly, we build weighted networks by calculating the embeddedness $\mathbf{E}_t$ and dispersion $\mathbf{D}_t$ between each pair of nodes to measure the strength of the relationships at each timestamp t. Then we construct a node strength matrix in which each element represents the connection strength of a pair of nodes. Combining the structural information at previous timestamp, the nonnegative matrix factorization method is used to detect the community structure for the dynamic networks. Finally, the experiments for two synthetic networks show that when considering the previous information, the accuracy of our algorithm improve 0.3425, 0.5191 for the first synthetic networks. For the second synthetic networks, the accuracy of our algorithm is also improved. Furthermore, we compare the other two algorithms, the results show that our algorithms perform better than other algorithms on the both synthetic networks. Our work may be helpful for providing a new perspective that we detect community structures for dynamic networks.

Key words: Community structure, Dynamic networks, Nonnegative matrix factorization.
PACS: 89.65.-s, 87.23.Ge, 89.75.Fb

Email address: liujg004@ustc.edu.cn (Jian-Guo Liu*).

1 Introduction

Many social networks exhibit the property of containing community structure[1–6], which has attracted much attention in the past several decades. Girvan and Newman[2] highlight the property of community structure, in which network nodes are joined together in tightly knit groups, between which there are only looser connections. The recent years have seen tremendous efforts for discovering community structure in static networks[7]. Meanwhile, we know that the networks are evolving over time owing to the changes of human behaviors. Detecting community structure for dynamic networks has been receiving increasing attention [8–13]. We classify the available algorithms for dynamic community detection into two categories: non-evolutionary based and evolutionary based approaches. Non-evolutionary algorithms first discover the local communities in the network at each timestamp and then analyze the involving relationship of the communities at successive timestamp. For instance, Kumar *et al.* [14] classified firstly members of network into groups and then studied the dynamics for social networks. Sun *et al.* [15] proposed the parameter-free GraphScope to discover communities in dynamic networks. Asur *et al.* [16]characterized the evolution events for communities in dynamic networks. Tang *et al.* [17] introduced a spectral clustering framework to discover communities and evolving rules. However, they ignore the connection between subsequent time steps, resulting in undesirable communities. To overcome the problem, evolutionary clustering [18,19] uses temporal smoothness for dynamic community detection, which balances the communities obtained in networks at two subsequent timestamps. FacetNet (a Framework for Analyzing Communities and EvoluTions in dynamic NETworks)[20] employs a stochastic block model to obtain communities at each time step, then uses a probabilistic model based on the Dirichlet distribution to capture evolution communities. The particle-and-density algorithm [21] (Kim-Han) makes use of the network topological structure, where the density-based algorithm is used to obtain local clustering at each time step, and uses the nano-community to obtain dynamic communties. Chi *et al.* [18] extended spectral clustering to trace the dynamics of communities, where two frameworks for evolutionary spectral clustering were proposed. Dynamic multiobjective genetic algorithm (DYNMOGA) [22] addresses dynamic community detection by reformulating the temporal smoothness as a multi-objective optimization problem, where the algorithm maximizes the snapshot cost and minimizes the temporal cost simultaneously. Furthermore, Liu *et al.* [23] developed the evolutionary co-clustering algorithms for dynamic networks. Mucha *et al.*[1] provides a framework for the study of community structure in a very general setting, covering networks that evolve over time by extending the popular modularity function (EMF) for community detection, and by adapting its implicit null model to fit a multislice networks. Each slice has an adjacency matrix describing connections between nodes belonging to the previously considered slice. This concept also includes interslice couplings which connects a node of a specific slice S_α to its copy in another slice S_β. The mathematical formulation of multiplex networks has been recently developed through many works [24,25]. For instance, a comprehensive formalism to deal with multiplex systems is proposed, and a number of metrics to characterize multiplex systems with respect to node degree, link overlap, node participation to different layers, clustering coefficient and eigenvector centrality are provided [24].

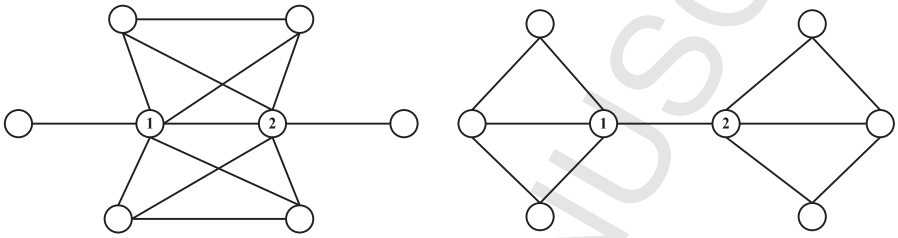

Fig. 1. The illustrated networks with strong connection strength of the relationships between nodes and weak connection strength. From the figure, we can find that in (a) the common neighbors of node 1 and 2 are connected, which shows that they have stronger connection strength. However, there is not common neighbors between node 1 and 2 in (b), that can be considered as weaker connection strength between node 1 and 2.

Although great efforts have been made for detecting dynamic communities, there are still many unsolved problems. For instance, how to make full use of the network structure information [26], and improve the accuracy of algorithms to detect community structure. We know that there is stronger strength or weaker strength between people in real social networks [27,28]. As shown in Fig.1, we can find that in (a) the common neighbors of node 1 and 2 are connected, that can be considered as strong connection strength. However, there is not common neighbors between node 1 and 2, that can be considered as weak connection strength in (b). How to measure the strength of the relationship plays an important role in the study of network structure. We assume that the strong relationship people tend to join in the same community. Inspired by this idea, we propose a community structure algorithm using the nonnegative matrix factorization[29,30]. Firstly, embeddedness [31] and dispersion [32] index are used to describe the strong or weak relationship between people. Based on these, we construct the node strength matrix at each timestamp. Then combining the previous structural information, the nonnegative matrix factorization method is used to detect the community structures for the dynamic networks at each timestamp[33]. Finally, we explore experiments in two synthetic networks. Experimental results show that our proposed algorithm has higher accuracy.

2 The community detection method

2.1 Preliminaries

We model the dynamic networks as a sequence of networks $g = \{G_1, G_2, \cdots, G_n\}$, where $G_t = (V_t, E_t)$ denotes a network at a timestamp t. And V_t, E_t are the sets of nodes and links of network at timestamp t respectively. A community in a static network G_t is a group of nodes $V_t^i \subseteq V_t$ having a high density of links inside the group, and a low density of links with the remaining nodes V_t/V_t^i. Let C_t represents a community for the graph G_t. Suppose there are N_c^t communities detected at the tth timestamp, denoted by $\{C_t^1, C_t^2, \cdots, C_t^{N_c^t}\}$, where the ith community could be denoted as $C_t^i = (V_t^i, E_t^i)$ and $V_t^i \subseteq V_t$, $E_t^i \subseteq E_t$. Without loss of generality, we assume that all of the networks in g have the same node set, i.e. $G_t = (V, E_t)$. The dynamic network g can be represented by a 3-dimensional adjacency matrix $\mathbf{A} = (a_{ijt})_{N\times N\times T}$, where N is the number of nodes and $a_{ijt} = 1$ if the i-th node is connected to the j-th node in G_t, 0 otherwise. Actually, $\mathbf{A}_t$ is symmetrical when G_t is undirected. In this paper, we assume that all the networks are undirected and unweighted.

2.2 Embeddedness and Dispersion

In this paper, we introduce two measurements, that is embeddedness [31] and dispersion [32], to describe the connection strength of each pair nodes. Embeddedness is described as the measurement in structural analyses for identifying close relation, capturing how much the two friends social circles overlap[34,35]. The embeddedness measurement $\mathbf{E}_t(u_t, v_t)$ for a pair nodes u_t, v_t at timestamp t can be expressed as,

$$\mathbf{E}_t(u_t, v_t) = \frac{n_{u_t v_t}}{(k_{u_t} - 1)(k_{v_t} - 1) - n_{u_t v_t}}, \tag{1}$$

where u_t is a node at timestamp t, v_t is a neighbor of u_t, k_{u_t} is the degree of the node u_t and $n_{u_t v_t}$ is the number of common neighbors for the node u_t and v_t.

The links to a person's relationship partner or other closest friends may have lower embeddedness, but they will often involve mutual neighbors from several different foci, reflecting the fact that the social orbits of these close friends are not bounded within any one focus, for example, a husband knows several of his wifes co-workers, family members, and former classmates, even though these people belong to different foci and do not know each other. The measure of dispersion looks not just at the number of mutual friends of two people, but also at the network structure on these mutual friends; roughly, a link between two people has high dispersion when their mutual friends are not well connected to one another. We now formulate a sequence of definitions that capture this idea of dispersion. To begin with, we define $C_{u_t v_t}$ to be the set of common neighbors of u_t and v_t. We define the dispersion of the $u_t - v_t$ link, $\mathbf{D}_t(u_t, v_t)$, to be the sum of all pairwise distances between vertices in $C_{u_t v_t}$ at timestamp t, that is,

$$\mathbf{D}_t(u_t, v_t) = \sum_{i,j \in C_{u_t v_t}} d_{v_t}(i, j), \tag{2}$$

where d_{v_t} is a distance function on the nodes of $C_{u_t v_t}$. We define $d_{v_t}(i, j)$ to be the function equal to 1 when i and j are not directly linked and also have no common neighbors in G_t other than u_t and v_t, and equal to 0 otherwise.

2.3 Method

Considering the temporal cost, we proposed a new method of detecting community structure for dynamic networks based on the nonnegative matrix factorization. Firstly, we present nonnegative matrix factorization method [36–38] to detect community structure. Nonnegative matrix factorization (NMF) is efficient for mining patterns in networks. NMF aims at learning the representation parts of the original data by approximating the target matrix into the product of two low-rank matrices. Specifically, given an $N \times M$ matrix $\mathbf{X}$, NMF decomposes $\mathbf{X}$ into two non-negative matrices $\mathbf{R}_{N\times r}$ and $\mathbf{F}_{r\times M}$ such that

$$\mathbf{X} \approx \mathbf{RF}, s.t.\mathbf{R} \geq 0, \mathbf{F} \geq 0. \tag{3}$$

In the community structure detection, N is the number of nodes and r is the number of communities in a network. Among those cost functions, the Least Squares Error (LSE) and Kullback-Leibler (KL) divergence[39] are more frequently selected. Recently, one of the recent NMF models is Symmetric Nonnegative Matrix Factorization (SNMF) model. Though there are several variations of NMF for detecting community structure, almost all of them can be solved by Multiplicative Update Rules. In this paper, we use the LSE cost function as following,

$$J(\mathbf{X}, \mathbf{R}, \mathbf{F}) = \|\mathbf{X} - \mathbf{RF}^T\|^2_{\mathbf{R}}. \tag{4}$$

And then we can fix $\mathbf{F}$ (or $\mathbf{R}$) and apply the gradient descent method to minimize the LSE in order to get the update rule of $\mathbf{R}$ (or $\mathbf{F}$) in Multiplicative Update Rules.

$$
\begin{aligned}
f_{ib} &:= f_{ib}\frac{(\mathbf{XR})_{ib}}{(\mathbf{FR}^T\mathbf{R})_{ib}};\\
r_{ib} &:= r_{ib}\frac{(\mathbf{XF})_{ib}}{(\mathbf{RF}^T\mathbf{F})_{ib}}.
\end{aligned}
\tag{5}
$$

To use the topological information adequately, we use the connection strength matrix to detect communities. According to Eq.(1)-(2), we construct the connection strength matrices $\mathbf{E}_t$ and $\mathbf{D}_t$ at each timestamp t. Furthermore, to discover the local clusters at timestamp t based on the temporal smoothness framework, we take into account both G_t and G_{t-1} via a linear function, which is defined as

$$
\mathbf{X}_t^* = \alpha\mathbf{X}_{t-1} + (1-\alpha)\mathbf{X}_t, \tag{6}
$$

where parameter α controls the relevant importance of history information. The underlying assumption is that if a group of nodes whose connectivity is strong in both G_{t-1} and G_t, then they are very likely to be a local cluster at time t. Usually, α=0.8. We set as $\mathbf{X}_t = \mathbf{E}_t$ or $\mathbf{X}_t = \mathbf{D}_t$. By nonnegative matrix factorization for the matrix $\mathbf{X}_t^*$, we can detect the communities of vertices at timestamp t. We denote the methods as T-E-NMF (combining time information, Embeddedness matrix with nonnegative matrix factorization), T-D-NMF (combining time information, Dispersion matrix with nonnegative matrix factorization) respectively. The whole process is described in **Algorithm 1** for the T-E-NMF. Meanwhile, we also investigate the performance of our algorithm without considering the previous information, which denotes the algorithm as E-NMF (Embeddedness matrix with nonnegative matrix factorization) and D-NMF (Dispersion matrix with nonnegative matrix factorization).

Algorithm 1 Pseudo-code of T-E-NMF Method

G is a set of the initial network

Input:

G: Dynamic networks;

N_c^t: The number of communities at each timestamp t;

Output: local clusters $\{C_t^1, C_t^2, \ldots, C_t^{N_c^t}\}$

1: For each time step t, construct $\mathbf{E}_t$ according to Eq.(1);

2: Construct the partial matrix $\mathbf{X}_t^*$ according to Eq.(6),

set as $\mathbf{X}_t = \mathbf{E}_t$;

3: Use NMF to obtain the communities for each node

at timestamp t according to Eqs.(4) and (5);

4: **Return**

For the algorithm T-D-NMF, we replace the formula of the second step on this process with $\mathbf{X}_t = \mathbf{D}_t$, and construct $\mathbf{D}_t$ according to Eq.(2).

Due to the influence of the non-negative matrix factorization method, our algorithm needs to know the number of community structures in advance. In this paper, we make the experiments on two synthetic networks which have known community structure information of each node at each timestamp.

3 The experimental analysis

3.1 Data sets

In this paper, we introduce two synthetic networks to evaluate the performance of the T-E-NMF and T-D-NMF algorithms.

The first artificial dataset is introduced [20], where each network consists of 128 nodes divided into 4 communities with 32 nodes each. Every node has a fixed average degree, and connects z nodes in other communities. We set the average degree of networks as 20, and generate the dynamic networks by varying parameter z. In order to introduce dynamics in the networks, $\delta\%$ of nodes are moved among communities. In our work, 10% of nodes in each community are randomly selected and assigned to the others at each time step.

The second synthetic dataset has been generated by taking into account some main events that may characterize the evolution of dynamic networks [40–42]. To this end we assumed four types of events, as introduced by Greene *et al.*[42]. The events are the following:

Birth and death: From the second time step on, 10% of new communities are created by removing nodes from other existing communities, and randomly removing 10% of the existing communities.

Expansion and contraction: 10% of communities are randomly selected and expanded or contracted by 25% of their size. When expanded, the new nodes are chosen at random from the other communities.

Intermittent communities: 10% of communities from the first time step are hidden.

Merging and splitting: At each time step, 10% of communities are split, 10% of communities are chosen, and couples of communities are merged.

To simplify the problem, in our experiments, we only produced data sets for two time steps and analyzed the accuracy of the algorithms on the second time slice.

3.2 Measurement

In our experiments, the normalized mutual information (NMI) [43,44] is used as the standard to evaluate the community structure detection performance. The value can be formulated as follows:

$$\mathrm{NMI}(M_1, M_2) = \frac{\sum_{s=1}^{N_c}\sum_{t=1}^{N_c} N_{xy}\log\frac{N_{xy}N}{N_x N_y}}{\sqrt{\sum_{x=1}^{N_c} N_x \log\frac{N_x}{N}\sum_{y=1}^{N_c} N_y \log\frac{N_y}{N}}}, \tag{7}$$

where M_1 is the real community label and M_2 is the computed community label, N_c is the community number, N is the number of nodes, N_{xy} is the number of nodes in the real community x that are assigned to the computed community y, N_x is the number of nodes in the real community x and N_y is the number of nodes in computed community y, $\log$ is the natural logarithm. NMI takes its maximum value of 1 if partition M_1 identical to partition M_2, and NMI=0 if two partitions are statistically independent.

Furthermore, we define the improvement rate η of the algorithm to evaluate the performance of our algorithm.

$$\eta = \frac{a-b}{b}, \tag{8}$$

where a is the result obtained by our algorithms, b is the result obtained by the comparing algorithms which could be E-NMF, D-NMF, DYNMOGA and EMF.

3.3 The experimental results

Firstly, we compare with the results of our algorithms whether or not considering the previous information. The results are shown in Fig.2 and Fig.3 for the synthetic networks with $z = 20$ and $\delta\% = 10\%$.

As shown in Fig. 2, the NMI for the T-E-NMF is 0.8149 that is higher than NMI of E-NMF, and the NMI for the T-D-NMF is 0.7769 which is higher than D-EMF. The values of NMI for E-NMF and D-NMF are 0.6070, 0.5114 respectively.

Then, we analyze the performance of our algorithm on the second synthetic networks. We generate four synthetic data sets for the four different types of events, for 2 time steps. The parameters to the generator have been set such that each network is constituted by 1000 nodes having mean degree of 15 and maximum degree 50, number of communities between 20 and 50, and mixing parameter (percentage of edges between communities) 0.2. Figure 3 depicts the NMIs on the four different data sets. One can find that when considering the previous information, our algorithms achieve well results. In the case of birthdeath, expand, hide and mergesplit,

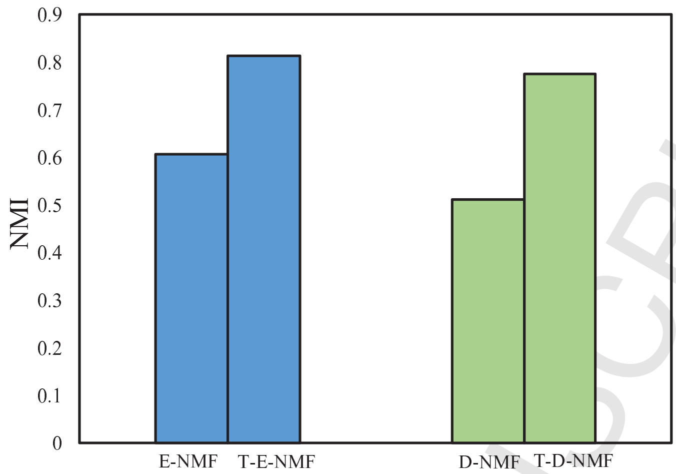

Fig. 2. The NMI for T-E-NMF, T-D-NMF, E-NMF and D-NMF on the first synthetic networks, where the algorithms T-E-NMF and T-D-NMF consider the previous structure information by calculating the strength between nodes with embeddedness and dispersion. One can find that the NMI of the T-E-NMF and T-D-NMF are higher than E-NMF and D-NMF respectively. This shows that the algorithms T-E-NMF and T-D-NMF are more accurate when considering the previous information.

the NMI values obtained by both T-E-NMF and T-D-NMF algorithm are greater than the corresponding E-NMF and D-NMF algorithm. In four cases, the values of NMI are close to 1 for the algorithm T-E-NMF and T-D-NMF, which shows that our algorithms can accurately detect the community structure of the networks.

Furthermore, in order to evaluate the performance of the T-E-NMF and T-D-NMF algorithms, two well-known algorithms are adapted for comparison, including DYNMOGA [22] and EMF[1]. The DYNMOGA and EMF are the well known algorithms and perform well for evolutionary clustering. Two datasets are used to validate the performance, including two synthetic dynamic networks. The synthetic networks have various evolution events, which testify whether the compared algorithms can accurately discover different evolution communities. The results are shown in Fig. 4 and 5.

Figure 4 reports the NMI of various algorithms for the first synthetic dynamic networks. It demonstrates that T-E-NMF and T-D-NMF achieve the best performance in the synthetic dynamic networks. In details, we can find that the values of NMI maximally research 0.8149, 0.7769 for T-E-NMF and T-D-NMF. However the values of NMI are 0.7037, 0.3047 for EMF, DYNMOGA respectively.

Figure 5 depicts the NMI on the four different data sets. The figure clearly shows that both T-E-NMF and T-D-NMF outperform DYNMOGA and EMF in all four evolution events, and our algorithms achieve the best performance in the four evolution events. In particular, DYNMOGA reaches a value not more than 0.90, while T-E-NMF or T-D-NMF obtain the values no less than 0.91 for four cases, especially the NMI value of algorithm T-E-NMF is 1 for the case of mergesplit. The

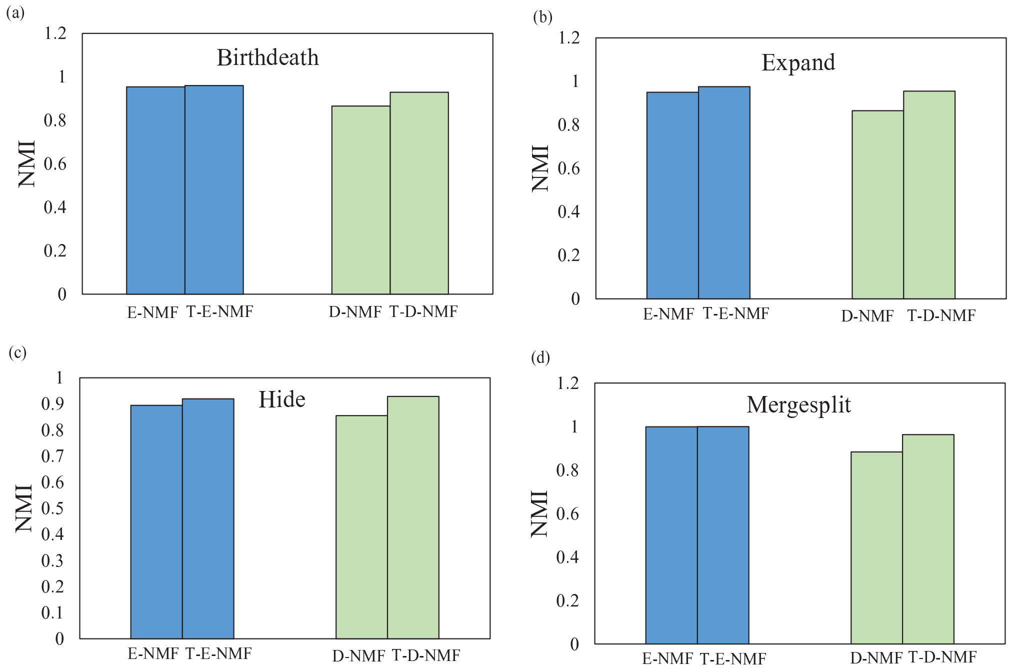

Fig. 3. Comparison between T-E-NMF and E-NMF, T-D-NMF and D-NMF on the second synthetic networks: (a) Birth and death, (b)Expansion and contraction, (c)Intermittent communities and (d) Merging and splitting, from which one can find that the T-E-NMF and T-D-NMF perform well for the synthetic data sets.

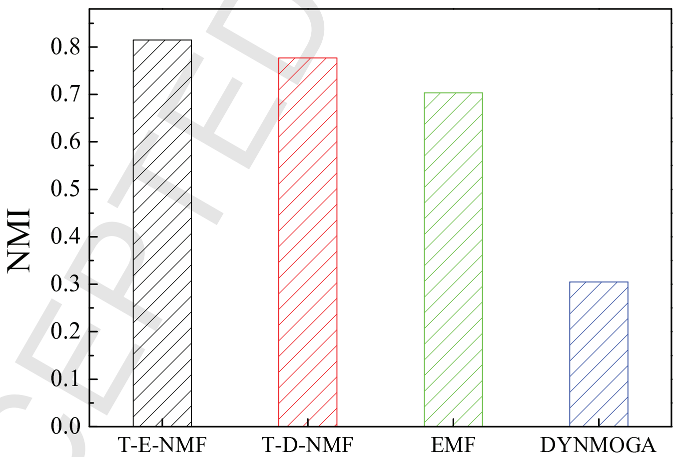

Fig. 4. The NMI for DYNMOGA, EMF, T-E-NMF and T-D-NMF on the first synthetic dataset, from which one can find that the values of NMI for T-E-NMF and T-D-NMF are 0.81, 0.78 respectively, which are higher than that of other algorithms. This shows that the presented algorithm works better for the synthetic dataset.

EMF method performs not well on the synthetic networks, and the values of NMI range from 0.4 to 0.65 in the four events.

Further, to illustrate the accuracy of our algorithm, we calculated the rate of improvement η compared to each of the other algorithms. The results are shown in

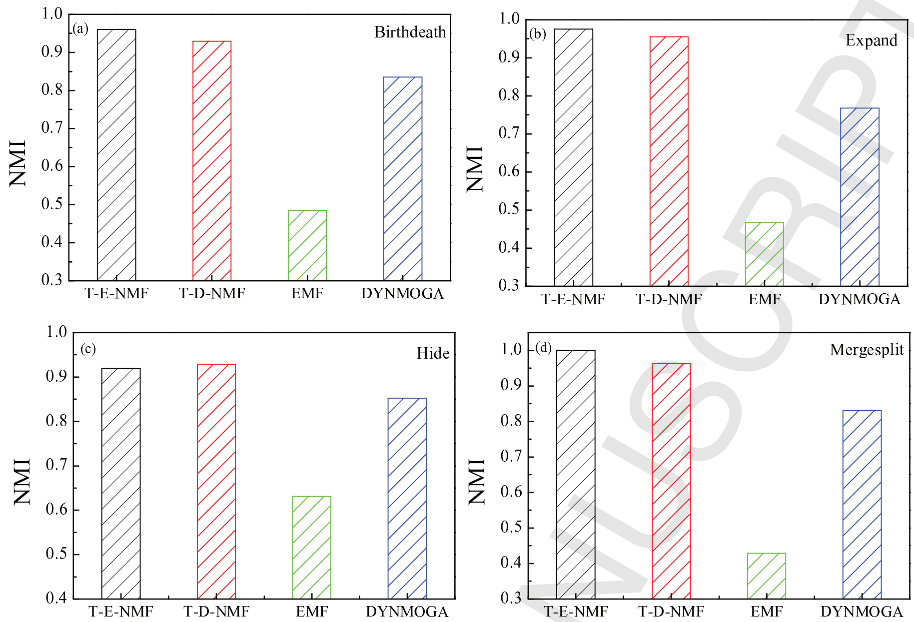

Fig. 5. Comparison among DYNMOGA, EMF, T-E-NMF and T-D-NMF on synthetic dataset: (a) Birth and death, (b)Expansion and contraction, (c) Intermittent communities and (d) Merging and splitting, from which one can find that the values of NMI for T-E-NMF and T-D-NMF on the four events are no less than 0.91, which suggests that the T-E-NMF and T-D-NMF perform well for the synthetic data sets.

Table 1 and 2.

From Table 1, we can see that the T-E-NMF algorithm is improved by 0.3425, 0.1581 compared with the E-NMF, EMF algorithm respectively. In particular, the T-E-NMF algorithm is 1.6741 times more accuracy than the DYNMOGA algorithm. Meanwhile, the η is 0.5191, 0.1040 for T-D-NMF compared with D-NMF, EMF. The η will be 1.5493 times to DYNMOGA for the T-D-NMF.

Table 1
The rate of improvement η for the other algorithms on the first synthetic networks.

η	T-E-NMF	T-D-NMF
E-NMF	0.3425	-
D-NMF	-	0.5191
EMF	0.1581	0.1040
DYNMOGA	1.6741	1.5493

In the second synthetic networks, from the Table 2, we can find that in the case of birthdeath, the algorithm T-E-NMF is improved by 0.0063, 0.9785 and 0.1450 respectively compared with algorithm E-NMF, EMF and algorithm DYNMOGA, the NMI for the T-D-NMF increases by 0.0737, 0.9144 and 0.1127 compared with the algorithm D-NMF, EMF and DYNMOGA respectively. Similarity in other three cases, the values of the rate η are 0.0272, 1.0852 and 0.2698 in case of expand, 0.0278, 0.4564 and 0.0788 of hide, 0.0012, 1.3307 and 0.2047 of mergesplit for the T-E-NMF. Meanwhile, for the T-D-NMF, the improvement rate η has also more than 0, especially the improvement rate η is 1.2439 comparing with the EMF in the mergesplit. All of this shows that our algorithm has more accuracy than other algorithms.

Table 2
The rate of improvement η for the other algorithms on the second synthetic networks.

	birthdeath	expand	hide	mergesplit
T-E-NMF				
E-NMF	0.0063	0.0272	0.0278	0.0012
EMF	0.9785	1.0852	0.4564	1.3307
DYNMOGA	0.1450	0.2698	0.0788	0.2047
T-D-NMF				
D-NMF	0.0737	0.1043	0.0862	0.0897
EMF	0.9144	1.0423	0.4709	1.2439
DYNMOGA	0.1127	0.2437	0.0896	0.1598

4 Conclusion and discussions

In this paper, we mainly consider the connection strength of human relationship, and propose a community structure partition algorithm based on nonnegative matrix factorization for the dynamic networks. We assume that the strong connection strength people tend to join in the same community. Firstly, we used the embeddedness and dispersion to describe the connection strength of pair nodes. Then, we constructed the node strength matrix $\mathbf{E}_t$, $\mathbf{D}_t$ at each timestamp t. The nonnegative matrix factorization methods (T-E-NMF and T-D-NMF) for dynamic community detection were proposed by incorporating a partial information into NMF. Finally, we do experiments to evaluate our algorithm performance on the synthetic networks. We first compare the algorithms whether or not considering the previous structure information. The experimental results indicate that, the values of the normalized mutual information (NMI) obtained by our algorithm are improved 0.3425, 0.5191 on the first synthetic networks, and improved a little on the second synthetic networks for T-E-NMF and T-D-NMF. Then, we compared with other two algorithms (DYNMOGA and EMF), the results show that our algorithms perform better than other algorithms on the both synthetic networks.

We propose a detecting community structure algorithm for dynamic networks, however, there are following problems to be resolved. The number of communities should be previously given, considering the characteristics of real networks, the number of communities is unknown, how to determine exactly how many communities in the networks should be addressed[45]. After detecting the communities of nodes, tracing the community evolution is a challenge work, using the Markov process to describe the evolution of community structure[46] is an important method for this problem.

5 Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Grant Nos. 61773248, 71771152).

References

[1] P.J. Mucha, T. Richardson, K. Macon, M.A. Porter, J. Onnela, Community structure in time-dependent, multiscale, and multiplex networks, Science 328(5980) (2010) 876-878.

[2] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA 99(12) (2002) 7821-7826.

[3] A. Clauset, M.E.J. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E 70(6) (2004) 066111.

[4] M.E.J. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. USA 103(23) (2006) 8577-8582.

[5] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, Nature 435(7043) (2005) 814-818.

[6] Y. Pan, D.H. Li, J.G. Liu, J.Z. Liang. Detecting community structure in complex networks via node similarity, Physica A 389(14) (2010) 2849-2857.

[7] S. Fortunato, Community detection in graphs, Phys. Rep. 486(3-5) (2010) 75-174.

[8] P. Holme, J. Saramäki, Temporal networks, Phys. Rep. 519(3) (2012) 97-125.

[9] C. Tantipathananandh, T. Berger-Wolf, D. Kempe, A framework for community identification in dynamic social networks, Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2007, pp. 717-726.

[10] G. Palla, A.L. Barabási, T. Vicsek, Quantifying social group evolution, Nature 446(7136) (2007) 664-667.

[11] L. Tang, H. Liu, J. Zhang, Z. Nazeri, Community evolution in dynamic multi-mode networks, Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2008, pp. 677-685.

[12] K. Yang, Q. Guo, S.N. Li, J.T. Han, J.G. Liu, Evolution properties of the community members for dynamic networks, Phys. Lett. A 381 (11) (2017) 970-975.

[13] Z.K. Zhang, C. Liu, X.X. Zhan, X. Lu, C.X. Zhang, Y.C. Zhang, Dynamics of information diffusion and its applications on complex networks, Phys. Rep. 651 (2016) 1-34.

[14] R. Kumar, J. Novak, A. Tomkins, Structure and evolution of online social networks, Link Mining: Models, Algorithms, and Applications, Springer New York, 2010, pp. 337-357.

[15] J. Sun, C. Faloutsos, S. Papadimitriou, C. Faloutsos, Graphscope: parameter-free mining of large time-evolving graphs, Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2007, pp. 687-696.

[16] S. Asur, S. Parthasarathy, D. Ucar, An event-based framework for characterizing the evolutionary behavior of interaction graphs, ACM Transactions on Knowledge Discovery from Data (TKDD) 3(4) (2009) 16.

[17] L. Tang, H. Liu, J. Zhang, Identifying evolving groups in dynamic multimode networks, IEEE Transactions on Knowledge and Data Engineering 24(1) (2012) 72-85.

[18] Y. Chi, X. Song, D. Zhou, K. Hino, B.L. Tseng, On evolutionary spectral clustering, ACM Transactions on Knowledge Discovery from Data (TKDD) 3(4) (2009) 17.

[19] D. Chakrabarti, R. Kumar, A. Tomkins, Evolutionary clustering, Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 554-560.

[20] Y. R. Lin, Y. Chi, S. Zhu, H. Sundaram, B.L. Tseng, Analyzing communities and their evolutions in dynamic social networks, ACM Transactions on Knowledge Discovery from Data (TKDD) 3(2) (2009) 8.

[21] M.S. Kim, J. Han, A particle-and-density based evolutionary clustering method for dynamic networks, Proceedings of the VLDB Endowment 2(1) (2009) 622-633.

[22] F. Folino, C. Pizzuti, An evolutionary multiobjective approach for community discovery in dynamic networks, IEEE Transactions on Knowledge and Data Engineering 26(8) (2014) 1838-1852.

[23] S. Ji, W. Zhang, J. Liu, A sparsity-inducing formulation for evolutionary co-clustering, Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 334-342.

[24] F. Battiston, V. Nicosia, V. Latora, Structural measures for multiplex networks, Phys. Rev. E 89(3) (2014) 032804.

[25] L. Solá, M. Romance, R. Criado, J. Flores, A. García del Amo, S. Boccaletti, Eigenvector centrality of nodes in multiplex networks, Chaos 23(3) (2013) 033131.

[26] J.G. Liu, J.H. Lin, Q. Guo, T. Zhou, Locating influential nodes via dynamics-sensitive centrality, Sci. Rep. 6 (2016) 21380.

[27] A. Petróczi, T. Nepusz, F. Bazsó, Measuring tie-strength in virtual social networks, Connections 27(2) (2007) 39-52.

[28] X.Q. Cheng, F.X. Ren, H.W. Shen, Z.K. Zhang, T. Zhou, Bridgeness: a local index on edge significance in maintaining global connectivity, J. Stat. Mech. Theory Exp. 2010(10) (2010) P10011.

[29] L. Yu, J. Huang, G. Zhou, C. Liu, Z.K. Zhang, TIIREC: A tensor approach for tag-driven item recommendation with sparse user generated content, Information Sciences 411 (2017) 122-135.

[30] L. Yu, C. Liu, Z.K. Zhang, Multi-linear interactive matrix factorization, Knowledge-Based Systems 85 (2015) 307-315.

[31] S. Aral, D. Walker, Tie strength, embeddedness, and social influence: A large-scale networked experiment, Manag. Sci. 60(6) (2014) 1352-1370.

[32] L. Backstrom, J. Kleinberg, Romantic partnerships and the dispersion of social ties: a network analysis of relationship status on facebook, Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, 2014, pp. 831-841.

[33] X. Ma, D. Dong, Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks, IEEE Transactions on Knowledge and Data Engineering 29(5) (2017) 1045-1058.

[34] D.H. Felmlee, No couple is an island: A social network perspective on dyadic stability, Soc. Forces 79(4) (2001) 1259-1287.

[35] M. Kalmijn. Shared friendship networks and the life course: An analysis of survey data on married and cohabiting couples. Soc. Networks 25(3) (2003) 231-249.

[36] F. Wang, T. Li, X. Wang, S. Zhu, C. Ding, Community discovery using nonnegative matrix factorization, Data Min. Knowl. Discov. 22(3) (2011) 493-521.

[37] L.Y. Tang, S.N. Li, J.H. Lin, Q. Guo, J.G. Liu, Community structure detection based on the neighbor node degree information, Internat. J. Modern Phys. C 27 (04) (2016) 1650046.

[38] J. Yang, J. Leskovec, Overlapping community detection at scale: a nonnegative matrix factorization approach, Proceedings of the sixth ACM International Conference on Web Search and Data Mining, ACM, 2013, pp. 587-596.

[39] D.D. Lee, H.S. Seung, Algorithms for non-negative matrix factorization, Advances in Neural Information Processing Systems (2001) 556-562.

[40] S. Asur, S. Parthasarathy, D. Ucar, An event-based framework for characterizing the evolutionary behavior of interaction graphs, ACM Transactions on Knowledge Discovery from Data (TKDD) 3(4) (2009) 16.

[41] C. Tantipathananandh, T. Berger-Wolf, D. Kempe, A framework for community identification in dynamic social networks. Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2007, pp. 717-726.

[42] D. Greene, D. Doyle, P. Cunningham, Tracking the evolution of communities in dynamic social networks, Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on IEEE, 2010, pp. 176-183.

[43] A. Lancichinetti, S. Fortunato, Community detection algorithms: a comparative analysis, Phys. Rev. E 80(5) (2009) 056117.

[44] L. Danon, A. Diaz-Guilera, J. Duch, A. Arenas, Comparing community structure identification, J. Stat. Mech. Theory Exp. 2005(09) (2005) P09008.

[45] M.E.J. Newman, G. Reinert, Estimating the number of communities in a network, Phys. Rev. Let. 117(7) (2016) 078301.

[46] L. Hou, X. Pan, Q. Guo, J.G. Liu, Memory effect of the online user preference, Sci. Rep. 4 (2014) 6560.