

Noise-tolerance community detection and evolution in dynamic social networks

Li Wang · Jiang Wang · Yuanjun Bi · Weili Wu ·
Wen Xu · Biao Lian

Published online: 25 February 2014
© Springer Science+Business Media New York 2014

Abstract Dynamic complex social network is always mixed with noisy data and abnormal events always influence the network. It is important to track dynamic community evolution and discover the abnormal events for understanding real world. In this paper, we propose a novel algorithm Noise-Tolerance Community Detection (NTCD) to discover dynamic community structure that is based on historical information and current information. An updated algorithm is introduced to help find the community structure snapshot at each time step. One evaluation method based on structure and connection degree is proposed to measure the community similarity. Based on this evaluation, the latent community evolution can be tracked and abnormal events can be gotten. Experiments on different real datasets show that NTCD not only eliminates the

L. Wang (✉) · J. Wang · B. Lian
College of Computer Science and Technology, Taiyuan University of Technology,
Shanxi 030024, China
e-mail: l_lwang@126.com; wangli@tyut.edu.cn

J. Wang
e-mail: jokerjiang@163.com

B. Lian
e-mail: lianbiao@126.com

Y. Bi · W. Wu · W. Xu
Department of Computer Science, University of Texas at Dallas,
Richardson, TX 75080, USA
e-mail: yuanjun.bi@utdallas.edu

W. Wu
e-mail: weiliwu@utdallas.edu

W. Xu
e-mail: wenxu@utdallas.edu

influence of noisy data but also discovers the real community structure and abnormal events.

Keywords Dynamic community detection · Abnormal events detection · Noise-tolerance

1 Introduction

We can observe the common principles in nature and human society that individuals are always inclined to those similar one. Based on this principle, some sub groups can be formed. If represented by graph model, those sub graphs are called communities by Newman (2004), where nodes are connected tightly within each community and the connection is loose outside the community. Finding community is very important for understanding the characteristics of complex network, discovering latent topology, predicting network evolution and so on.

Due to its significant theoretical research value and widely applications, many researchers from computer science, sociology, mathematics, physics science are focusing on community study. Many novel models are proposed and new applications are exploited. However, most current work focuses on finding static communities. They ignore the change of observed data over time and the models they discovered always do not obtain the topology changing process and lose some latent information.

It is necessary to delve into the dynamic aspects of network behavior, yet it would not be feasible without the data to support such explicitly dynamic analysis. With the development of computer networks and wide applications of social network software, the human society and computer networks fuse much more than before. Many applications on network, such as email, blog, facebook, instant communication, mobile network, sensor networks and so on, supply facile abundant source of dynamic datasets for studying and finding more detailed dynamic topology information in computer network-based social networks.

2 Related work

Recently, some work on identifying dynamic community is published. Their intentions of researching dynamic community can be classified into four types: finding stable core community by analyzing its changes; finding key attributes and predicting future evolution that cannot be observed from snapshot dataset; eliminating the influence of noisy data and getting individual characteristics; inspecting abnormal events. The main algorithms are dynamic spectral, dynamic clustering, graph evolutionary and so on.

Most methods focus on evolution clustering and optimization models in which the quality of snapshot community and the community evolution process are estimated by different weight parameters and both factors are summed in the final evaluation. Chakrabarti et al. (2006) firstly brought forward evolutionary clustering method and its optimization model. They considered that there is relationship between time t and $t+1$ and its changing should be smooth. Chi et al. (2007) firstly put forward

evolutionary spectral clustering algorithm. He extended similarity computing methods and utilized graph cut to measure community structure and community evolution. [Lin et al. \(2008\)](#) made use of nonnegative matrix factorization, based on Markov probability model and dirichlet distribution to build dynamic community framework *Facenet*. She aimed at overlapped community and proposed soft community and soft modularity models. [Chayant et al. \(2007\)](#) studied dynamic community from a new perspective. He enumerated the different situations of community dynamic change and set up different cost evaluation methods that include individual cost, group cost, color cost and built optimization model for dynamic community detecting. [Tang et al. \(2008\)](#) proposed a spectral algorithm to model dynamic community evolution.

There are some graph evolutionary algorithms related with dynamic community detection. [Kumar et al. \(2003\)](#) focused on blogosphere. By building blog graph, they researched blog community evolution and burst law based on the change of in-degree, out-degree, strongly connected components and so on. [Leskovec et al. \(2005\)](#) analyzed graph evolution models in various fields and proposed generators that produce graphs exhibiting the discovered patterns.

Except evolutionary clustering, graph evolution and optimization models, some methods are put forward recently. [Zhao et al. \(2008\)](#) advanced a general framework for heterogenic network to forecast community members. They firstly compressed the objects and relationships that are in the same period and built vector-based heterogenic networks. Then they respectively extracted the characteristics of network snapshots and network sequence, built snapshot-based feature and delta-based feature, set up multi level classification and regression model and made some post processes. [Palla et al. \(2007\)](#) analyzed a co-authorship network and a mobile network, which both networks are dynamic, by using the clique percolation method (CPM). They found that larger scale communities will stand longer than small communities. [Toyoda and Kitsuregawa \(2003\)](#) analyzed community changes such as emerge, dissolve, grow and shrink and so on. They built a set of metrics to qualify such changes and set up the evolution model of web community. [Spiliopoulou et al. \(2006\)](#) proposed a framework *MONIC* to monitor cluster transitions over time, which includes a set of external transitions such as survive, split, disappear. Also, in their framework, the transactions among different clusters and a set of internal transitions such as size and location transitions are modeled to capture the changes within a community. [Asur et al. \(2007\)](#) defined a set of events on both individual and community to model community evolution. They defined a set of metrics to calculate the individual and community stability, sociability, influence, popularity and so on. [Sun et al. \(2007\)](#) put forward a parameter-free algorithm GraphScope that extracts community and detect community change from time-evolving graphs by Minimum Description Length (MDL) principle. [Mei et al. \(2005\)](#) extracted latent themes from texts and mined temporal text on topic evolving graphs. [Leskovec et al. \(2005\)](#), [Leskovec et al. \(2008\)](#) studied graph evolution characteristics such as average degree, distance between nodes, conductance, community profiles and so on. [Kumar et al. \(2006\)](#) divided social network into three classes: singletons, giant component and middle region and studied their evolution characteristics. [Falkowski et al. \(2006\)](#) studied social network evolution over time at sub graph level. [Sarkar and Moore \(2005\)](#) proposed to embed nodes to latent spaces and detect the node location change over time to analyze community evolution.

Gomez-Rodriguez and Leskovec (2013) developed an online algorithm for dynamic network inference that relies on stochastic convex optimization. Wang (2011) built self-organization communities for network management. Lancichinetti et al. (2011) proposed a method, order statistics local optimization method (OSLOM), for detecting clusters based on local optimization of a fitness function. Nguyen et al. (2011) focused on mobile social networks and put forward AFOCS to detect communities adaptively to help mobile applications.

Although there are many different models for dynamic community identification, the most common procedure is to divide dataset into different snapshots firstly, then discover communities at each snapshot and compare adjacent snapshot topologies to get the evolution information. How to choose proper time window to build snapshot? How to deal with data sparseness, noise data and missing data at snapshot? These challenges sometimes can influence the result accuracy greatly. Some researchers took historic information into account to filter noise, but they did not consider the time difference of historic information and change degree. Integrating history information can make complementary for data sparse and noise. But how to integrate to get better solution is still a challenge.

In this paper, we put forward a novel algorithm Noise-Tolerance Community Detection (NTCD) to identify dynamic communities that is based on historical information and snapshot information. We focus on weighted graph because dynamic community detection is more valuable in dynamic interaction social networks than in social relationship networks that is more stable and unweighted graph is a special weighted graph.

We utilize the algorithm Wang et al. (2013) that is an improvement of Blondel et al. (2008) to find the community structure snapshot at each time step. One evaluation formula is proposed to detect unusual change of community. The main contributions of this paper are threefold:

- (1) We build an updated method to integrate historical information and current information for snapshot networks.
- (2) We build a new algorithm for measuring the similarity of community topology that is based on community structure and community weight.
- (3) We put forward a simple measurement for detecting urgent change of community.

All of these help to overcome the influence of noise and find the change points of community topology.

3 Preliminaries and basic definitions

3.1 Preliminaries

According to social experiences, some assumptions and preliminaries are built as the following.

1. Both historical information and observed snapshot information are important for discovering communities. But their importance is different and the historical infor-

mation effect will decrease with time. Furthermore, the change degree may have influence on the value of historical information.

2. Generally the objects evolution is smooth. So historical information will help when observed data is not complete or noise involved. But the supplementary is local sensitive and usually the adjacent former data is more useful than others.
3. Noise data is useless and sometimes lead to wrong solution. In dynamic community analysis, absence of observed data, improper data are all noise data. Noise data may make some abnormal change. Abnormal change may caused by some urgent events. The difference between noise-caused abnormal and event-caused abnormal is the change sequence. Noise-caused abnormal always is very short and event-caused abnormal will stay for a long time. We can identify abnormal events by analyzing dynamic communities.

According to the above preliminaries, we proposed NTCD algorithm to detect communities for unweighted and weighted networks.

3.2 Updating method for snapshot graph

OG_t shows the observed social network at time t and $OG_t = \{V_t, E_t\}$. V_t is the node set that can be observed at time t and E_t is the edge set that records the interaction or relationship status happening at time t . At each time step there is a snapshot graph OG_t . These snapshot graphs constitutes a snapshot sequence $\{OG_1, OG_2, \dots, OG_t, OG_{t+1}, \dots\}$.

We set the updated edge value as w . Generally when the change between adjacent snapshots is big, the effect of historical information is small, vice versa. To regularize the value, we build the formula (1) and the updated edge weight is as formula (2).

$$P = \frac{e_i - e_{i-1}}{\max\{e_i, e_{i-1}\}} \quad (1)$$

$$W = \alpha * e^{-p-1} e_{i-1} + (1 - \alpha * e^{-p-1}) e_i \quad (2)$$

e_{i-1} , e_i represent the edge weight at time $t-1$ and t . α is a constant. It is related to Δt and their correlation is negative. That is, bigger is Δt , smaller is α . Because the interval for snapshots usually is same, α is an invariable value in one background. $p \in [-1, 1]$, it shows the directed conversion between adjacent snapshots. $\alpha * e^{-p-1} \in [\alpha * e^{-2}, \alpha] \subset (0, \alpha]$. It shows the historical information effect based on the degradation with time and the weight variety. It is bigger than 0 and smaller than α , which means not only the historical information indeed influences the current observed data, but also the effect of historical information is limited in some range.

4 Noise-tolerance community detection in dynamic social networks

When detecting static communities, a proper quality measurement is important. Modularity shows the difference between community topology and stochastic topology with same degree distribution. Because “Modularity represented one of the first attempts to

achieve a first principle understanding of the clustering problem, it is by far the most used and best known quality function” Fortunato (2010). The algorithm proposed by Blondel et al. (2008) is one of the most widely used methods in modularity based algorithms because of its performance, especially in the big data era. Some improvement on this algorithm is proposed in Wang et al. (2013). So we use the improvement algorithm to detect community at each time step.

For community sequence, we calculate the community difference between adjacent timestamps to decide whether there are change points or urgent group events.

The detailed information of three main sub procedures in NTCD is the following.

4.1 Updating rules for dynamic graphs OG_t

If we observe new nodes at one time, add them to OG_t . If one node disappears temporally or dies, we do not delete it from OG_t immediately, because its historic information still influences the network topology. If we observe some new interactions, we can update the edge weight by formula 1 and 2. If some edges already exist, but at this time we can not observe them, when we update OG_t we still update its weight because its weight will decrease with time.

4.2 Algorithm of partition community at one time t

There are some important definitions in the improvement community detection algorithm Wang et al. (2013).

Definition 1 *Overlapping degree (OV)*. OV denotes the overlapping degree of two nodes neighbors, which shows the connection strength between nodes.

There are two different measurement methods of overlapping degree. One is based on the weight sum of edges between nodes and shared neighborhoods. The other one is based on the size of shared neighborhood. So we give two definitions to calculate the nodes connection strength. One is OV_{weight} that is based on the weight. For nodes i and j , their OV_{weight} values can be calculated by the following formula:

$$OV_{weight}(i, j) = \frac{\sum_{p=1, w_p \in N_i \cap N_j}^k (weight(i, w_p) + weight(j, w_p))}{\sum_{p=1, w_p \in N_i \cup N_j}^k (weight(i, w_p) + weight(j, w_p))} \tag{3}$$

The other alternative one is OV_{node} that shows the ratio of shared neighborhood size to the size sum of both neighborhoods. We define it as follows:

$$OV_{node} = \frac{|N_i \cap N_j|}{|N_i \cup N_j|} \tag{4}$$

where N_i is the neighbor set of node i . $weight(i,j)$ is the weight value of edge (i,j) . $|R|$ is the node number of set R .

Definition 2 *Swing nodes.* Swing nodes are those nodes that have no clear community structure with their neighbors. That means the connection strength between them and their neighbors are all smaller than some threshold value,

$$OV(i, j) < \Theta, j \in Ni \quad (5)$$

where Θ is a threshold.

The improvement algorithm consists two iterative phases and one pre procedure for reordering nodes. If the $OV(i,j)$ is smaller than threshold Θ , move it to the delay node set, otherwise do it.

The detailed improved algorithm is in ref [Wang et al. \(2013\)](#).

4.3 Algorithm for measuring the similarity of community structures

When we get community snapshot sequence, we can analyze the community topology continuity of adjacent time steps. Computing community similarity is crucial for uncovering the evolution process.

We build two measurements for tracking community evolution. One is based on community structure and the other one is based on community connection degree. They can uncover the change on global topology and local structure respectively.

(1) Measurement on global topology

C_t and C_{t-1} respectively show community structure of time t and $t-1$. Because the number of communities in different time maybe different, when calculating community similarity we do not know which one community at time $t+1$ comes from that community at time t . We compare the whole set. Normalized mutual information (NMI) [Danon et al. \(2005\)](#) is a widely used measurement of similarity between two partitions based on information theory, which has been proved to be reliable. There may exist some communities without matching objects at another timestamp. We call these communities as instant communities and we set the similarity between instant communities and other communities as 0.

Just as the [3.1](#) preliminaries, from the evolution process we can distinguish the group events caused by noise and that by abnormal change. If one topology change happened and this change lasts for a period, this change may be caused by abnormal group events. If one structure change happened and the change only lasts for a very short time, it may be caused by noise data. So we set the following measurement for detecting abnormal change point.

If $NMI_{i-1} > NMI_i$ && $NMI_{i+1} > NMI_i$, then the i th time is one change point and some new events or abnormal group event happened around at the i th time. NMI_i is the community similarity between the i th snapshot and $i + 1$ th snapshot.

(2) Measurement on local structure

In social networks, events may cause more interaction in some communities without global community topology change. Such local interaction may lead tighter and more

static connection relationship. They also play an important role in tracking community evolution. The measurement formula is the following.

$$\Delta_{weight}(C_i^{t-1}, C_i^t) = \left| \frac{\sum_{a,b \in C_i^{t-1}} w(a,b)}{\|C_i^{t-1}.edge\|} - \frac{\sum_{a,b \in C_i^t} w(a,b)}{\|C_i^t.edge\|} \right| \tag{6}$$

C_i^{t-1} is the i^{th} community at the $t-1$ snapshot. $C_i^{t-1}.edge, C_i^t.edge$ are edge set of C_i^{t-1}, C_i^t respectively. $w(a,b)$ is the weight of edge (a,b) .

5 Experiments and evaluations

Although some related work is proposed, dynamic community detection is still in its infancy [Fortunato \(2010\)](#) and standard evaluation method is lacked. One popular evaluation way is to compare the identified community structure with real structure; especially it is very useful to analyze the match between identified structures and its semantic content or events. We did experiments on different datasets. One is Enron email dataset, which is a kind of online social network and records the exchange email information between staffs in Enron Corporation. Its size is medium and it has 158 nodes and 600 thousand edges. The other one is from Sina weibo, which is a Chinese social media. This dataset records the online interaction and retweets between users. It has 1.77 million users and 23.755 million retweets. Its size is some large and the data distribution is very sparse. Moreover, there were abnormal events happened in Enron dataset. We can verify our result based on the real world.

We detect communities on these dynamic networks. OSLOM [Lancichinetti et al. \(2011\)](#) is one noise-tolerance community detection method and its goal is somehow similar with our method. So we compare our NTCD and OSLOM on different datasets.

5.1 Experiments on Enron email datasets

- (1) Email relationship can show the organization structure of one company. Email communication can show the business status to some extent. We divided Enron email dataset according to month, computed the snapshot community topology and made a comparison of NMI from January to December in 2001. Because in Enron corporation, during 2001 some important events have happened that made some changes on their organization structure. Furthermore in order to investigate the influence of parameter α , we did different community detection computing for different parameter α . Parameter α decides the effect of historical information, we set it respectively as 0.3, 0.5 and 0.7.

The NMI from January to December in 2001 is as the Fig. 1.

In the Figure, we can see that there are great difference between the result of observed_based and that of NTCD. In observed_based curve, there are two pits, 4 and 11. In NTCD and OSLOM curves there are 3 pits, 5, 8 and 11. The historical information makes the curve more smooth and then more easy to detect the change

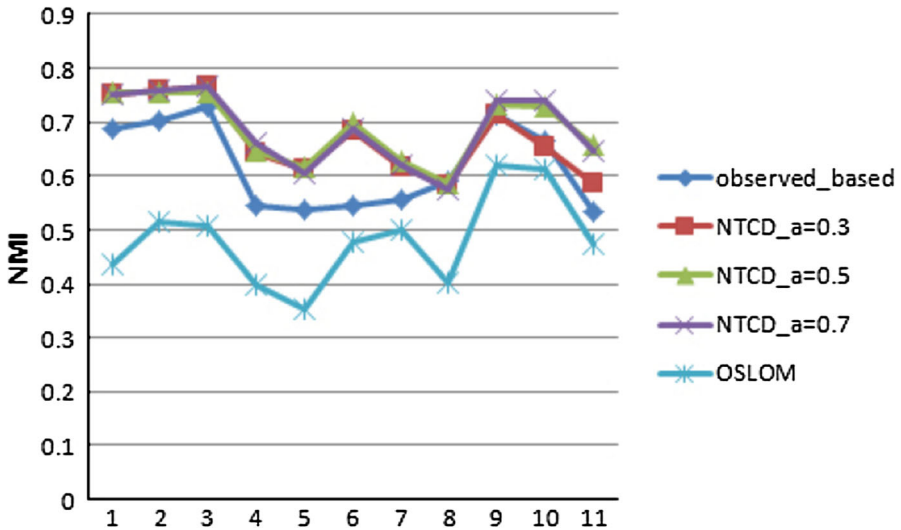


Fig. 1 NMI comparison from January to December of 2001 in Enron email dataset

points. So our NTCD strategy taking historical information into considered is more helpful than observed_based algorithm for tracking the community evolution.

But on the other hand, we also see that the results of different parameter α show almost same trend. The subtle distinguish is that after September, the NMI of NTCD_ $\alpha = 0.3$ and that of observed_based are more alike, which is because during those times there were some great changes on email relationship and the less historical information, the more change community topology. But whatever α is equal to, bigger or smaller than 0.5, these NTCD algorithms all show the same discovery that May, August, November are key points. So we can draw a conclusion that the NMI is insensitive to α if α is near 0.5 and 0.5 is a suitable value. So in the later of this paper we set α is 0.5.

Furthermore, to verify the result correctness, we traced to the real Enron scandal and found that some important events happened in those times.

In 6th May, the company *Off Wall Street* published an analysis report suggesting investors to sell Enron stock because of the lower and lower profit. Although there have been some queries from a few traditional media organizations, most reports were praise by that time.

In August, Jeffrey Skilling, the CEO of Enron Corporation unexpectedly resigned from Enron.

In the end of October, SEC (United States Securities and Exchange Commission) began the official investigation on Enron Coporation. And in 2nd Dec., Enron filed for bankruptcy protection in the Southern District of New York. In 3rd Dec., Enron reduced about 4,000 staff.

The three events are all very important for Enron. So NTCD helps identify the abnormal events.

In the other hand, OSLOM has the nearly same result as NTCD except its NMI values generally are smaller than NTCD.

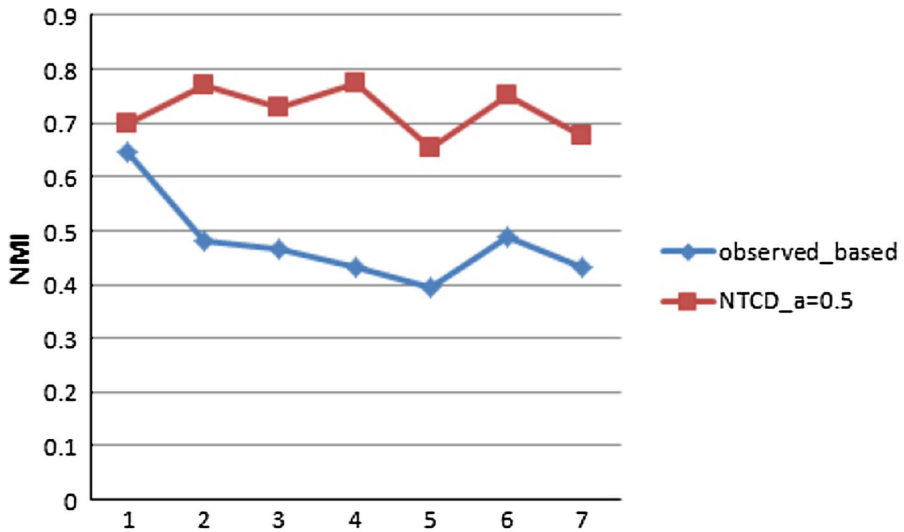


Fig. 2 NMI of week snapshot from November 6 to December 31 of 2001

Table 1 Performance comparison of NTCD and OSLOM

Month	NTCD	OSLOM
7	<1s	3s
8	<1s	4s
9	<1s	4s

- (2) If the time window is small, what result will be gotten? We also did some work on dynamic community analysis on weeks. We chose the email data from Nov.6 to Dec.31 of 2001 because there were some abnormal events happened. The NMI status is as the Fig. 2. From the Fig. 2, we can see all NTCD and observed_based algorithms show the 5th node is one abnormal change point. In real world, at that time Enron filed for bankruptcy protection and reduced the staff. So the two curves show the correct information. But NTCD shows the change more clearly than observed_based computing.

5.2 Performance analysis

In experiment on Enron dataset, OSLOM has the analogous result as NTCD. It also can identify the important change points that imply the abnormal events. But NTCD has the better performance than OSLOM.

The Table 1 shows some performance comparison of NTCD and OSLOM on computing dynamic communities in July, August and September of 2001. The experiment environment is same, double core CPU, memory 1,024 M, operation system is ubuntu 10.10.

We make a theoretical analysis on NTCD and it is $O(n^2)$.

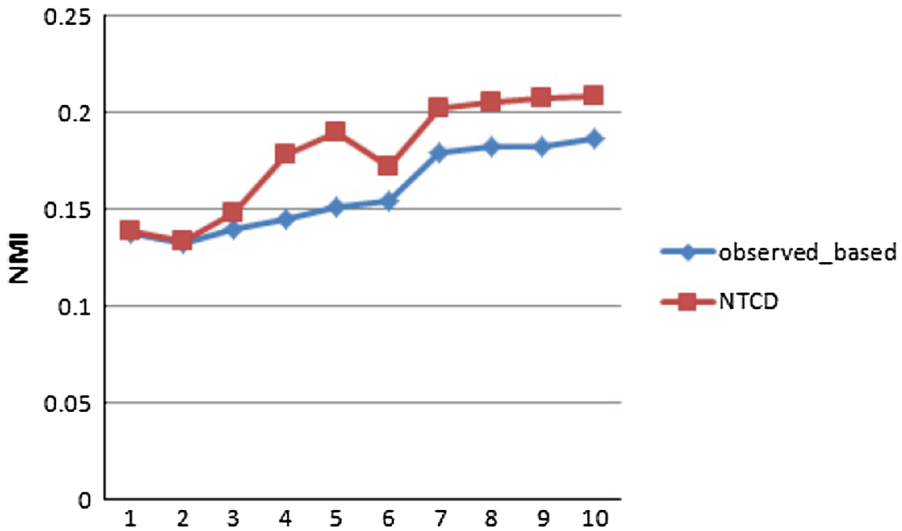


Fig. 3 NMI comparison of sina weibo dataset

5.3 Experiments on weibo dataset

Sina weibo is a big social media platform in China. We use the dataset that crawled by Tsinghua University (<http://arnetminer.org/Influencelocality>). The dataset records the retweet relationship among users. There are many complex factors that lead to retweet action. The interaction topology is more sparse and more dynamic. We did NTCD on this dataset to identify the latent community evolution trend.

We divided the data by month and computed the NMI from January to November in 2011. The NMI comparison figure is Fig. 3.

Firstly we checked the community modularity at each snapshot and found that they almost were bigger than 0.5. This means community topology exists in retweet network. Secondly we computed the NMI of adjacent communities and found their NMIs are all very small, which means the change between adjacent community topologies is some great. But as time went by and the retweet interaction increased, the NTCD and observed_based curve all show rising trend, which imply the similarity between adjacent community increases. But from NTCD curve there is a sudden change at 6th node, which means an abnormal event maybe happen during that time. But in the observed_based curve we cannot get the information of change point.

6 Conclusions

In this paper, we proposed a novel algorithm NTCD to identify dynamic communities in social networks. Differentiated from other algorithms, it uncovers community snapshots based on historic information and current observed network and discovers the community evolution process and abnormal events based on evaluation of the adjacent community structures. Experiments on different real datasets show that NTCD

not only eliminates the influence of noisy data but also discovers the real community structure and abnormal events, thus helps better understanding of real network structure and its evolution.

Acknowledgments Partially supported by the Major State Basic Research Development Program of China (Grant No. 2013CB329602), the International Collaborative Project of Shanxi Province, China (Grant No. 2011081034), the National Natural Science Foundation of China (Grant No. 61202215, 61100175, 61232010). China postdoctoral funding (Grant No. 2013M530738).

References

- Asur S, Parthasarathy S, Ucar D (2007) An event-based framework for characterizing the evolutionary behavior of interaction graphs. In: Proceedings of the 13th ACM SIGKDD Conference
- Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. *J Stat Mech* P10008
- Chakrabarti D, Kumar R, Tomkins A (2006) Evolutionary clustering. In: Proceedings of the 12th ACM SIGKDD Conference
- Chi Y, Song X, Zhou D, Hino K, Tseng BL (2007) Evolutionary spectral clustering by incorporating temporal smoothness. In: Proceedings of the 13th ACM SIGKDD Conference
- Danon L, Duch L, Guilera AD, Arenas A (2005) Comparing community structure identification. *J Stat Mech* 9:P09008
- Falkowski T, Bartelheimer J, Spiliopoulou M (2006) Mining and visualizing the evolution of subgroups in social networks. In: IEEE/WIC/ACM WI
- Fortunato S (2010) Community detection in graphs [J]. *Phys Rep* 486(3–5):75–174
- Gomez-Rodriguez M, Leskovec J (2013) Structure and dynamics of Info. pathways in online media, WSDM
- Kumar R, Novak J, Raghavan P, Tomkins A (2003) On the bursty evolution of blogspace. In: Proceedings of the 12th WWW Conference
- Kumar R, Novak J, Tomkins A (2006) Structure and evolution of online social networks. In: SIGKDD
- Lancichinetti A, Radicchi F, Ramasco JJ, Fortunato S (2011) Finding statistically significant communities in networks. *PLoS One* 6:e18961
- Leskovec J, Kleinberg J, Faloutsos C (2005) Graphs over time: desiccation laws, shrinking diameters and possible explanations. In: SIGKDD
- Leskovec J, Kleinberg J, Faloutsos C (2005) Graphs over time: densification laws, shrinking diameters and possible explanations. In: Proceedings of the 11th ACM SIGKDD Conference
- Leskovec J, Lang KJ, Dasgupta A, Mahoney MW (2008) Statistical properties of community structure in large social and information networks. In: ACM WWW
- Lin YR, Chi Y, Zhu S, Sundaram H, Tseng BL (2008) Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. WWW '08: Proceeding of the 17th international conference on World Wide Web New York, NY. ACM p. USA, pp 685–694
- Mei Q, Zhai C (2005) Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In: Proceedings of the 11th ACM SIGKDD Conference
- Newman MEJ (2004) Detecting community structure in networks. *Eur Phys J B* 38(2):321–330
- Nguyen NP, Dinh TN, Tokala S, Thai MT (2011) Overlapping communities in dynamic networks: their detection and mobile applications. In *MobiCom '11: Proceedings of the 17th annual international conference on mobile computing and networking*, pp 85–96
- Palla G, Barabasi AL, Vicsek T (2007) Quantifying social group evolution. *Nature* 446:664–667
- Sarkar P, Moore AW (2005) Dynamic social network analysis using latent space models. *SIGKDD Explor. Newsl.* 7(2)
- Spiliopoulou M, Ntoutsis I, Theodoridis Y, Schult R (2006) Monic: modeling and monitoring cluster transitions. In: Proceedings of the 12th ACM SIGKDD Conference
- Sun J, Faloutsos C, Papadimitriou S, Yu PS (2007) GraphScope: parameter-free mining of large time-evolving graphs. In: Proceedings of the 13th ACM SIGKDD Conference
- Tang L, Liu H, Zhang J, Nazeri Z (2008) Community evolution in dynamic multi-mode networks. In: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining Las Vegas, Nevada, USA, pp 677–685

- Tantipathananandh C, Berger-Wolf T, David Kempe A (2007) Framework for community identification in dynamic social networks. KDD, august. California, USA, pp 717–726
- Toyoda M, Kitsuregawa M, (2003) Extracting evolution of web communities from a series of web archives. In HYPERTEXT 03: Proceedings of the 14th ACM conference on hypertext and hypermedia
- Wang L (2011) SoFA : an expert-driven, self-organization peer-to-peer semantic communities for network resource management. *Expert Syst Appl* 38(1):94–105
- Wang L, Wang J, Shen H-W, Cheng X-Q (2013) Improvement on fast uncovering community algorithm. *Chinese phys B* 22(10):108903. doi:[10.1088/1674-1056/22/10/108903](https://doi.org/10.1088/1674-1056/22/10/108903)
- Zhao Q, Bhowmick SS, Zheng X, Kai Y (2008) Characterizing and predicting community members from evolutionary and heterogeneous networks. In: Proceeding of the 17th ACM conference on Information and knowledge management. Napa Valley, California, USA, pp 309–318