Community Detection Based on Links and Node Features in Social Networks

Fengli Zhang[1] , Jun Li[2], Feng Li[1], Min Xu[3],
Richard Xu[3] and Xiangjian He[3]

[1] Department of Automation, USTC, Hefei 230027,
Anhui, China
[2] School of Computer Science and Communication, AHPU,
WuHu 241000, Anhui, China
[3] School of Engineering and Communications, Faculty of Engineering and IT,
UTS, Sydney,Australia

zhangfli@mail.ustc.edu.cn, fli@ustc.edu.cn, beniciolee@126.com,
{Min.Xu, YiDa.Xu, Xiangjian.He}@uts.edu.au

Abstract. Community detection is a significant but challenging task in the field of social network analysis. Many effective methods have been proposed to solve this problem. However, most of them are mainly based on the topological structure or node attributes. In this paper, based on SPAEM [1], we propose a joint probabilistic model to detect community which combines node attributes and topological structure. In our model, we create a novel feature-based weighted network, within which each edge weight is represented by the node feature similarity between two nodes at the end of the edge. Then we fuse the original network and the created network with a parameter and employ expectation-maximization algorithm (EM) to identify a community. Experiments on a diverse set of data, collected from Facebook and Twitter, demonstrate that our algorithm has achieved promising results compared with other algorithms.

Keywords: Community Detection, Social Network, EM algorithm, Node Similarity

1 Introduction

Recently, with the exploration of Internet, social networking is becoming an increasingly significant application because it enables users from different places to connect with each other. Strong community structure [2] is one fundamental property of social network. A very meaningful task of social network analysis is community detection, which aims to partition the users who have denser connectivity into one cluster. Community detection is a powerful tool to understand the internal structure of the network, that is, how users interact with each other. If we use community as a basic unit when doing research on the social networks, the network can be simplified and compressed effectively so that we can mine useful information from complex network with acceptable computation cost. Community detection also has many other applications such as friend suggestion, product recommendation and link inference.

A number of algorithms have been proposed for community detection, such as G-N algorithm [2], Spectral Clustering [3], Neman's Mixture Model [4] and MMSB [5]. Most of these algorithms only focus on topological structure. To learn more about the related algorithms, we can see the recent surveys [6] [7]. However, in a real social network, there always exists link noise (incorrect links and missing links). The presence of link noise makes identifying community more difficult. For example, some nodes with no link or weak link but sharing fairly similar features may be grouped into distinct communities, which is unreasonable. Therefore, only considering the network links is not enough. In real life, people in one community not only have denser links but also more or less similarities among them. According to observation, we can take the node attributes into consideration to help alleviate the noise and strengthen the community signal.

In recent years, various algorithms have been proposed to combine the links and content for community detection. Zhu et al [8] introduce a method that jointly factorizes the content matrix and link matrix for a spectral clustering. Cohn and Hofmann [9] present a joint probabilistic model of document content and connectivity, an extension of PLSA [10, 11] and HITS [12, 13]. Erosheva et al [14] describe a mixed-membership model to analyze both the content of a document and its citation. Nallapati et al [15] present two different models called Pairwise-Link-LDA and Link-PLSA-LD. The former one combines LDA [16] and Mixed Membership Block Stochastic Model [5] and the other combines the LDA and PLSA models into a single graphical model. In [17], the objects such as photos and articles two users shared are regarded as edge content between them and then edge content is incorporated into the matrix factorization. In the article [18], the author presents CODICIL, a family of highly efficient graph simplification algorithms leveraging both content and graph topology to identify and retain important edges in a network. McAuley and Leskovec [19] try to automatically discover users' social circles fusing link and users' profile.

In this paper, we propose a joint probabilistic model of combining link and node features for community detection. In this work, we first build a SPAEM model only with the network links. Next, we create a new feature-based weighted network whose edge weight is the node feature similarity between two nodes. Then, we fuse the original network and the created network. If two nodes have a strong similarity, the original link between the two nodes will be strengthened, otherwise it will be weakened. How much the node features have impact on the original links can be determined by introducing a parameter. Finally, an expectation-maximization algorithm (EM) is employed for the optimization.

The rest of the paper is organized as follows. In section 2, we first review the SPAEM model. Then, how to create feature-based network has been discussed. Finally, we present the method of combining links and node features. In section 3, experimental results tested on different data sets are presented. Conclusions are drawn in section 4.

2 Our Method

In this section, we first introduce the SPAEM model and then create a new feature-based network. Next, we present a joint model combining link and node features. In the

following we assume that the network in the paper is undirected and unweighted. Let A denotes the adjacent matrix; $A_{ij}=1$ if there is a link between node i and j, otherwise $A_{ij}=0$.

2.1 SPAEM model

SPAEM [1] model regards community detection as a probabilistic inference problem. It utilizes the idea of the probabilistic latent semantic analysis [2] which is a powerful algorithm in text mining. Compared with other algorithms [20] [21], SPAEM model possesses the mathematical simplicity and hence is easy to understand.

We assume that $N_{(i)}$ denotes the set of the neighbors of node i. Suppose that: there is c latent communities to be detected; every node has probability π_r to fall in group r; community r selects node i with probability $\beta_{r,i}$ with constraint $\sum_{i=1}^{n}\beta_{ri}=1$.

The edge e_{ij} is generated by the following finite mixture model where the community r is latent variable.

(1) Select a community r with the probability π_r

(2) The node i with probability $\beta_{r,i}$ to be selected by Community r.

(3) The node j with probability $\beta_{r,j}$ to be selected by Community r.

Assume that community r selects node i and node j independently, the probability of choosing the node pair {i, j} is

$$P(e_{ij} \mid \pi,\beta)=P(\{i,j\} \mid \pi,\beta)=\sum_{r=1}^{c}\pi_r\beta_{r,i}\beta_{r,j} \tag{1}$$

High value of $P(e_{ij} \mid \pi,\beta)$ is regarded as a reliable edge. If there is a link in the node i and node j, they should have a high likelihood of joining in the same community, in other words, the nodes in the same community with high value of β should be connected.

The logarithm probability of network A under parameters π, β can be modeled as

$$\begin{aligned} L &= \ln P(A \mid \pi,\beta) \\ &= \sum_{i=1}^{n}\sum_{j:j\in N(i)} \ln P(e_{ij} \mid \pi,\beta) \\ &= \sum_{i=1}^{n}\sum_{j:j\in N(i)} \ln \sum_{r=1}^{c}\pi_r\beta_{r,i}\beta_{r,j} \end{aligned} \tag{2}$$

In order to optimize the value of parameters π, β, we maximize the logarithm probability by expectation-maximization (EM) algorithm [22].

E-step:

The posterior probability $P(g_{ij}=r \mid A,\pi,\beta)$, denoted by q_{ij}, then

$$
\begin{aligned}
q_{ij} &= P(g_{ij} = r \mid A, \pi, \beta) \\
&= \frac{P(g_{ij} = r, A \mid \pi, \beta)}{P(A \mid \pi, \beta)} \\
&= \frac{\pi_r \beta_{r,i} \beta_{r,j}}{\sum_{s=1}^{c} \pi_s \beta_{s,i} \beta_{s,j}}
\end{aligned}
\tag{3}
$$

M-step:
The expected logarithm probability of the network is

$$
\begin{aligned}
\vec{L} &= \sum_{i=1}^{n} \sum_{j:j \in N(i)} \sum_{r=1}^{c} q_{ij,r} \ln P(e_{ij}, g_{ij} = r \mid \pi, \beta) \\
&= \sum_{i=1}^{n} \sum_{j:j \in N(i)} \sum_{r=1}^{c} q_{ij,r} \ln \pi_r \beta_{r,i} \beta_{r,j}
\end{aligned}
\tag{4}
$$

By maximizing $\vec{L}$ we can get

$$
\pi_r = \frac{\sum_{i=1}^{n} \sum_{j:j \in N(i)} q_{ij,r}}{\sum_{i=1}^{n} \sum_{j:j \in N(i)} \sum_{s=1}^{c} q_{ij,s}}, \quad \beta_{r,i} = \frac{\sum_{j:j \in N(i)} q_{ij,r}}{\sum_{k=1}^{n} \sum_{j:j \in N(k)} q_{kj,r}}
\tag{5}
$$

2.2 Create Feature-based network

Assume that F_i denotes the set of the features of node i. The node feature similarity between node i and node j is defined by Jaccard coefficient. Next we create a link between node i and node j and take the value of the node feature similarity as the edge weight W_{ij} of node i and node j, that is

$$
W_{ij} = Jaccard(v_i, v_j) = \frac{|F_i \bigcap F_j|}{|F_i \bigcup F_j|}
\tag{6}
$$

If the value of feature similarity does not equal to 0, the two nodes form an edge with weight W_{ij}. Otherwise, there is no link between the nodes. Hence we get a weighted new network based on node features.

The new created network owns the same nodes with the original network. For any node i, it is impossible for community r to select the node with two possibilities at the same time. So in the new network, the probability that community r is selected can be

still denoted by π_r and the probability that community r select node i is still $\beta_{r,i}$. The probability of choosing the node pair {i, j} is the same as Equation (1).
For the new network, we use $N'_{(i)}$ to denote the neighbors of node i. Because the network is weighted, we replace A_{ij} with W_{ij}. The expected logarithm probability of the weighted network can be rewritten as.

$$\vec{L} = \sum_{i=1}^{n} \sum_{j:j\in N'(i)} \sum_{r=1}^{c} W_{ij} q_{ij,r} \ln \pi_r \beta_{r,i} \beta_{r,j} \tag{7}$$

The weighted network can be used to detect community just relying on the node features.

2.3 Combining Link and Node Features

The information of the social network cannot be fully utilized if just applying each separately. Similar to the work in [9] which is an influential algorithm of combining content and connectivity in text mining, it is reasonable to merge the two networks into a joint probabilistic model, therefore we propose maximizing the following expected logarithm probability with a parameter α.

$$\begin{aligned} \vec{L} = \alpha \sum_{i=1}^{n} \sum_{j:j\in N(i)} \sum_{r=1}^{c} q_{ij,r} \ln \pi_r \beta_{r,i} \beta_{r,j} + \\ (1-\alpha) \sum_{i=1}^{n} \sum_{j:j\in N'(i)} \sum_{r=1}^{c} W_{ij} q_{ij,r} \ln \pi_r \beta_{r,i} \beta_{r,j} \end{aligned} \tag{8}$$

In this model, the original links have limited effect when detecting communities. Even though the link between node i and node j is weak, if the two nodes have strong similarity, the link will be strengthened. Thus, they may form more reliable edge and have high probability of belonging to the same community.

The value of α depends on different applications, that is, the importance one assigns to predict links and node features. When detecting community, if we think the link is more important, the value of α can be set with a higher value.

Next what we do is to calculate q_{ij}, $\beta_{r,i}$ with EM algorithm. In E-step, we compute the posterior probability q_{ij}. In M-step, substitute q_{ij} into Equation (8) and optimize $\beta_{r,i}$ by maximizing $\vec{L}$. The posterior probability $P(g_{ij}=r \mid A, \pi, \beta)$, denoted by q_{ij}, can still be computed by equation (3).

Taking the constraints into consideration: $\sum_{r=1}^{n} \pi_r = 1, \sum_{i=1}^{n} \beta_{ri} = 1$, the Lagrange function is

$$
\begin{aligned}
D=&\alpha\sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{r=1}^{c}q_{ij,r}\ln\pi_r\beta_{r,i}\beta_{r,j}+\\
&(1-\alpha)\sum_{i=1}^{n}\sum_{j:j\in N1(i)}\sum_{r=1}^{c}W_{ij}q_{ij,r}\ln\pi_r\beta_{r,i}\beta_{r,j}+\\
&\lambda(\sum_{r=1}^{c}\pi_r-1)+\sum_{r=1}^{c}\gamma_r(\sum_{i=1}^{n}\beta_{r,i}-1)
\end{aligned}
\tag{9}
$$

where λ, γ_r are Lagrange multipliers. The derivatives of D are

$$
\begin{aligned}
\frac{\partial D}{\partial\pi_r}&=\alpha\sum_{i=1}^{n}\sum_{j:j\in N(i)}\frac{q_{ij,r}}{\pi_r}+(1-\alpha)\sum_{i=1}^{n}\sum_{j:j\in N1(i)}\frac{W_{ij}q_{ij,r}}{\pi_r}+\lambda\\
\frac{\partial D}{\partial\beta_{r,i}}&=\alpha\sum_{j:j\in N(i)}\frac{q_{ij,r}}{\beta_{r,i}}+(1-\alpha)\sum_{j:j\in N1(i)}\frac{W_{ij}q_{ij,r}}{\beta_{r,i}}+\gamma_r
\end{aligned}
\tag{10}
$$

Combine with the constraints $\sum_{r=1}^{n}\pi_r=1$, $\sum_{i=1}^{n}\beta_{ri}=1$, and let the derivatives of Equation (10) equal to 0. We can get as follows,

$$
\pi_r=\frac{\alpha\sum_{i=1}^{n}\sum_{j:j\in N(i)}q_{ij,r}+(1-\alpha)\sum_{i=1}^{n}\sum_{j:j\in N1(i)}W_{ij}q_{ij,r}}{\alpha\sum_{i=1}^{n}\sum_{j:j\in N(i)}\sum_{s=1}^{c}q_{ij,s}+(1-\alpha)\sum_{i=1}^{n}\sum_{j:j\in N1(i)}\sum_{s=1}^{c}W_{ij}q_{ij,s}}
\tag{11}
$$

$$
\beta_{r,i}=\frac{\alpha\sum_{j:j\in N(i)}q_{ij,r}+(1-\alpha)\sum_{j:j\in N1(i)}W_{ij}q_{ij,r}}{\alpha\sum_{k=1}^{n}\sum_{j:j\in N(k)}q_{kj,r}+(1-\alpha)\sum_{k=1}^{n}\sum_{j:j\in N1(k)}W_{ij}q_{kj,r}}
\tag{12}
$$

By iterating Equations (3), (11) and (12) until convergence, we can obtain q_{ij}, $\beta_{r,i}$.

The probability community s selects of node i is $u_{s,i}=\pi_s\beta_{s,i}$. If community r meets the following condition, the node i belongs to community r.

$$
r=\arg\max_s\{u_{s,i}=\pi_s\beta_{s,i},s=1,2,...,r\}
\tag{13}
$$

The algorithm can also be used to detect overlapping community. For node i, $r=\arg\max_s\{u_{s,i}=\pi_s\beta_{s,i},s=1,2,...,r\}$, if there exists another community s such that $\frac{u_{s,i}}{u_{r,i}}>\theta(0<\theta<1)$, node i also belongs to community s.

3 Experiment

In this section, experiments on a small real data set are firstly carried on in order to intuitively demonstrate the difference between our method and the existing methods. Then, we experiment on the public data sets, i.e. Facebook[1] and Twitter[2], to observe the effect of parameter α and how our algorithm outperforms compared to other methods.

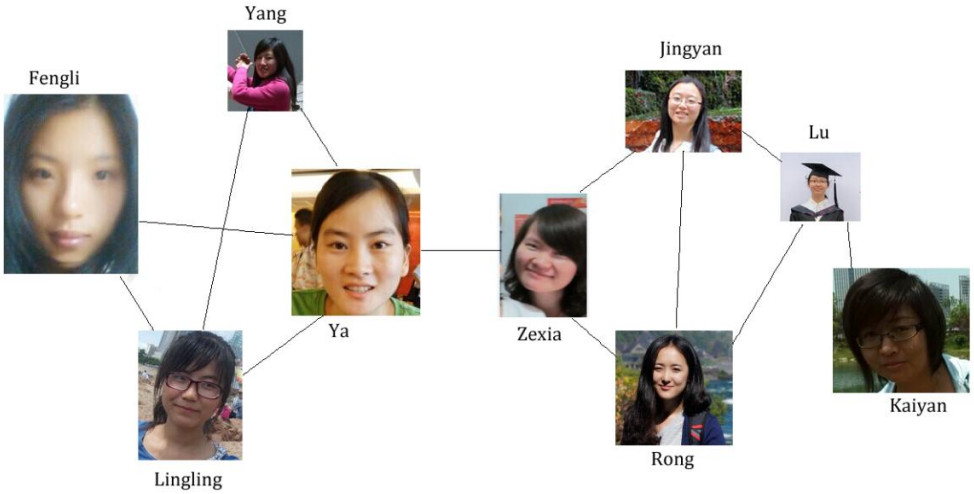


Fig.1. the connections among the nine students

3.1 Experiment on self-collected data set

In order to how node attributes and links affect the result and make the result of our algorithm more directed and visible, we apply the algorithm to a small real data set. The data set is collected by ourselves, which is about nine students in USTC: Lingling, Yang, Fengli, Ya, Zexia, Rong, Jingyan, Lu and Kaiyan. We investigate whether they have connection with each other when they just began their college life, as showed in Figure 1. The feature vectors of the nodes in Table 1 are their interests in music, dancing, reading, traveling and film. If one likes music, the value equals to 1. Otherwise, the value is set to 0. We group them into different communities and verify that whether the obtained result is consistent with the communities they formed in later days in their college lives.

We test our algorithm on the above data set. In the experiments, α is set to 1.0, 0 and 0.7 respectively. The results are showed as Figure 2. α=1 means that community detection is only based on the network link. As shown in the left of Figure 2, the nodes in one community are linked more densely. The center shows the results when node attributes are the only consideration. “Kaiyan” is grouped into the “green” community because of their very strong similarity. Considering both structure and node features, we set α to 0.7. In this case, “Kaiyan” is grouped into two communities at the same time. There is no doubt that “Kaiyan” should belong to the “red” community because of their

[1] http://snap.stanford.edu/data/egonets-Facebook.html.
[2] http://snap.stanford.edu/data/egonets-Twitter.html.

links. Besides, even though there is no link between “kaiyan” and “Lingling et al, they have strong feature similarity. The node features strengthen the links between them. So it is reasonable for “Kaiyan” grouped into the “green” community. In fact, in the following years in USTC Kaiyan usually does some extracurricular activities with “green” group, and attends classes together with the other group. So Kaiyan connects with both the groups and should be assigned to the two communities simultaneously. Apparently, the result of α=0.7 agrees better with the reality.

Table 1. the feature vectors of the nine students

	Lingling	Yang	Fengli	Ya	Zexia	Rong	Jingyan	Lu	Kaiyan
music	1	0	0	0	0	0	0	0	1
dancing	0	0	1	0	0	0	0	0	0
reading	1	1	1	1	0	0	1	0	1
traveling	1	1	1	1	0	0	0	0	1
film	0	0	0	0	1	1	1	1	0

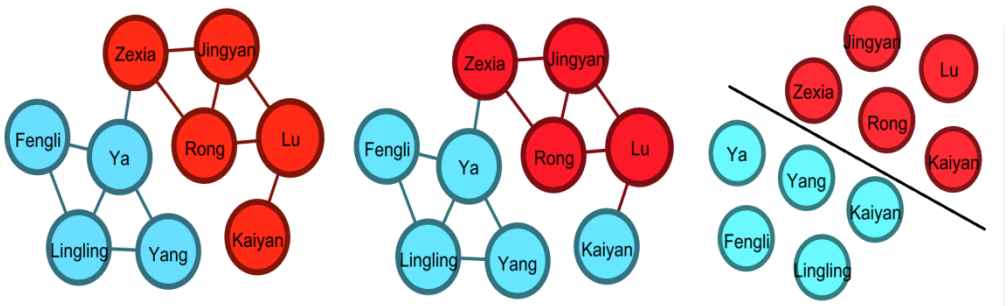


Fig.2. Experiment on small real data. The value of α from left to right is 1.0, 0 and 0.7

3.2 Experiments on public data set

In this sub-section, we first introduce the criterion to evaluate the quality of the detected results. Then we compare our method with other methods on the date sets, i.e. Facebook and Twitter according to the evaluation criterion. Facebook data was collected from survey participants using this Face app[3]. Twitter data was crawled from public sources. Both the data set includes node features, ground-truth circles and networks.

3.2.1 Evaluation criterion

To evaluate our algorithm, we maximize the consistency between the detected communities $\mathbb{C}=\{C_1...C_k\}$ and the ground-truth communities $\overline{\mathbb{C}}=\{\overline{C}_1...\overline{C}_k\}$.

The F-score of C on $\overline{C}$ is denoted as follows:

[3] https://www.facebook.com/apps/application.php?id=201704403232744.

$$\mathrm{F}(C,\bar{C}) = 2 \cdot \frac{precision(C,\bar{C}) \cdot recall(C,\bar{C})}{precision(C,\bar{C}) + recall(C,\bar{C})} \tag{14}$$

Precision and recall are defined as

$$precision(C,\bar{C}) = \frac{|C \cap \bar{C}|}{|C|}, recall(C,\bar{C}) = \frac{|C \cap \bar{C}|}{|\bar{C}|} \tag{15}$$

For each detected community C, we compute its F-score on $\bar{\mathbb{C}}$.

$$\mathrm{F}(C,\bar{\mathbb{C}}) = \max_{\bar{C} \in \bar{\mathbb{C}}} F(C,\bar{C}) \tag{16}$$

Then the final F-score of $\mathbb{C}$ on $\bar{\mathbb{C}}$ is:

$$\mathrm{F}(\mathbb{C},\bar{\mathbb{C}}) = \sum_{C \in \mathbb{C}} \frac{|C|}{|N|} F(C,\bar{\mathbb{C}}) \tag{17}$$

where N denotes the set of the nodes.

The higher value of $\mathrm{F}(\mathbb{C},\bar{\mathbb{C}})$ denotes the detected communities are closer to the ground-truth. We use it to measure the quality of all the algorithm in the following experiments.

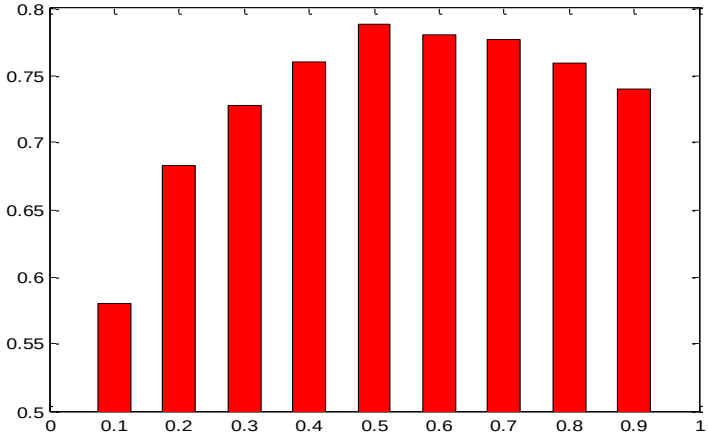
3.2.2 Effect of the parameter α

The value of α is decided experimentally, which depends on different data sets. In this sub-section, we track how the quality of detected communities changes as the value of α varies from 0.1 to 0.9. Figure 3 shows the results of different α on Facebook and Twitter. For Facebook, the best quality is achieved when α=0.5, while for Twitter, when α=0.7, F-score is the highest. The weight value is determined by nodes and links' importance, which varies for different applications. Therefore, for different data sets, we can adjust the value of α to obtain the best result.

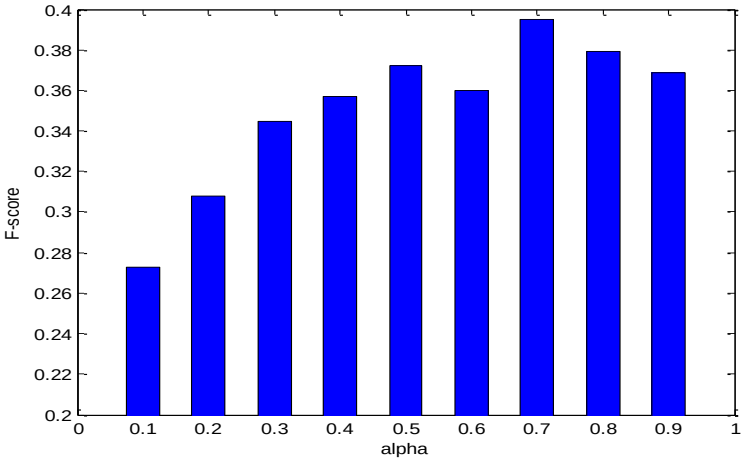
3.2.3 Comparison with other methods

In this section, we compare our method with MMSB [5], SPAEM [1], K-means, and MaAuley and Leskovec's algorithm (MLA) [19]. MMSB and SPAEM only focus on network links; K-means is a classical algorithm that considers node features only; MLA is a new algorithm to discover social circles combining links and node features. We apply these algorithms on the data set of Facebook and Twitter. In the experiment, we set α of our method to 0.5 and 0.7 respectively for Facebook and Twitter.

From the encouraging results (Figure 4 and Figure 5), our algorithm outperforms the other four methods significantly.



(a) Varying α on Facebook



(b) Varying α on Twitter

Fig. 3 effects of varying α

4 Conclusion

In this paper, we propose an algorithm combining the links and node features. In the algorithm, we create a new feature-based network and fuse it with the original network with a parameter to alleviate the noise and strengthen the community signal. Experimental results show that our method outperforms state-of-the-art methods in clustering quality. For the future work, first, we plan to improve computing efficiency of our algorithm to adapt to the large scale networks. Then, we will try other algorithms to

do optimization. The EM algorithm we adopt in our method is easy to fall into a local optimum. So we will utilize other algorithms to enhance the ability to search the global optimum.

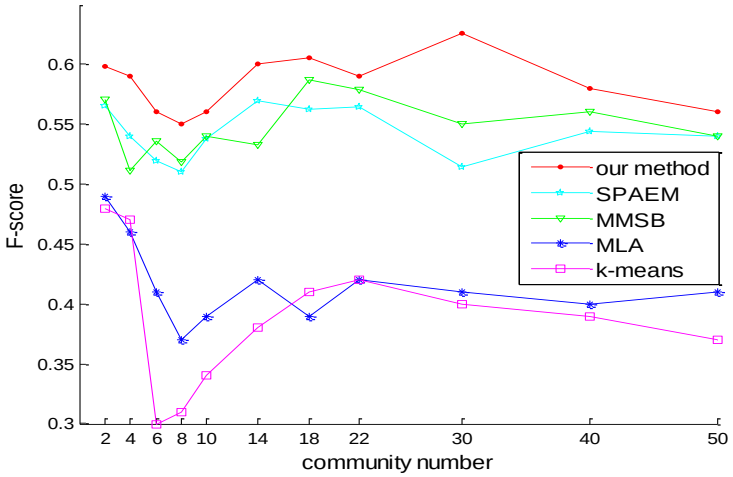


Fig. 4 Experiments on Facebook

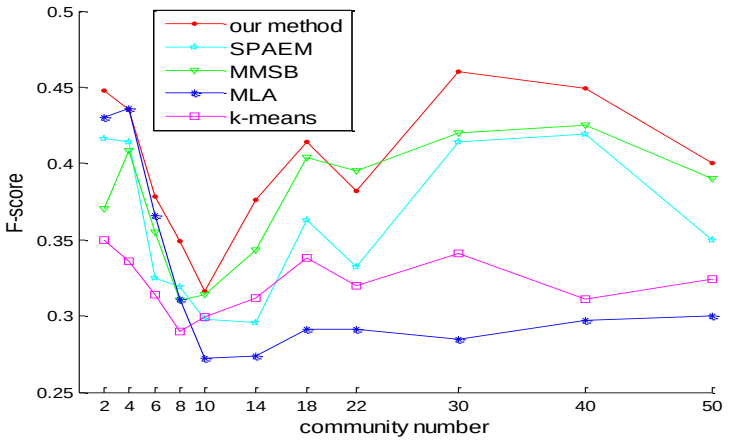


Fig.5 Experiments on Twitter

References

1. Ren, Wei, et al. "Simple probabilistic algorithm for detecting community structure." Physical Review E 79.3 (2009): 036111.

2. Girvan, Michelle, and Mark EJ Newman. "Community structure in social and biological networks." Proceedings of the National Academy of Sciences99.12 (2002): 7821-7826.
3. Pothen, Alex, Horst D. Simon, and Kang-Pu Liou. "Partitioning sparse matrices with eigenvectors of graphs." SIAM Journal on Matrix Analysis and Applications 11.3 (1990): 430-452.
4. Newman, Mark EJ, and Elizabeth A. Leicht. "Mixture models and exploratory analysis in networks." Proceedings of the National Academy of Sciences104.23 (2007): 9564-9569.
5. Airoldi, Edoardo M., et al. "Mixed membership stochastic blockmodels."Advances in Neural Information Processing Systems. 2009.
6. Fortunato, Santo. "Community detection in graphs." Physics Reports 486.3 (2010): 75-174.
7. Xie, Jierui, Stephen Kelley, and Boleslaw K. Szymanski. "Overlapping community detection in networks: The state-of-the-art and comparative study." ACM Computing Surveys (CSUR) 45.4 (2013): 43.
8. Zhu, Shenghuo, et al. "Combining content and link for classification using matrix factorization." Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2007.
9. Hofmann, David Cohn Thomas. "The missing link-a probabilistic model of document content and hypertext connectivity." Proceedings of the 2000 Conference on Advances in Neural Information Processing Systems. The MIT Press. 2001.
10. Deerwester, Scott C., et al. "Indexing by latent semantic analysis." JASIS41.6 (1990): 391-407.
11. Hofmann, Thomas. "Probabilistic latent semantic indexing." Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1999.
12. Cohn, David, and Huan Chang. "Learning to probabilistically identify authoritative documents." ICML. 2000.
13. Kleinberg, Jon M. "Authoritative sources in a hyperlinked environment."Journal of the ACM (JACM) 46.5 (1999): 604-632.
14. Erosheva, Elena, Stephen Fienberg, and John Lafferty. "Mixed-membership models of scientific publications." Proceedings of the National Academy of Sciences of the United States of America 101.Suppl 1 (2004): 5220-5227..
15. Nallapati, Ramesh M., et al. "Joint latent topic models for text and citations."Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008.
16. Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.
17. Qi, G-J., Charu C. Aggarwal, and Thomas Huang. "Community detection with edge content in social media networks." Data Engineering (ICDE), 2012 IEEE 28th International Conference on. IEEE, 2012.
18. Ruan, Yiye, David Fuhry, and Srinivasan Parthasarathy. "Efficient community detection in large networks using content and links." Proceedings of the 22nd international conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2013.
19. Leskovec, Jure, and Julian J. Mcauley. "Learning to discover social circles in ego networks." Advances in neural information processing systems. 2012.
20. Ramasco, José J., and Muhittin Mungan. "Inversion method for content-based networks." Physical Review E 77.3 (2008): 036122.
21. Vazquez, Alexei. "Population stratification using a statistical model on hypergraphs." Physical Review E 77.6 (2008): 066106.
22. Dempster, Arthur P., Nan M. Laird, and Donald B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm." Journal of the Royal Statistical Society. Series B (Methodological) (1977): 1-38.