CONTRASTIVE DEEP NONNEGATIVE MATRIX FACTORIZATION FOR COMMUNITY DETECTION

Yuecheng Li[1]　　*Jialong Chen*[1]　　*Chuan Chen*[1*]　　*Lei Yang*[1]　　*Zibin Zheng*[2]

[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2] School of Software Engineering, Sun Yat-sen University, Zhuhai, China

ABSTRACT

Recently, nonnegative matrix factorization (NMF) has been widely adopted for community detection, because of its better interpretability. However, the existing NMF-based methods have the following three problems: 1) they directly transform the original network into community membership space, so it is difficult for them to capture the hierarchical information; 2) they often only pay attention to the topology of the network and ignore its node attributes; 3) it is hard for them to learn the global structure information necessary for community detection. Therefore, we propose a new community detection algorithm, named **C**ontrastive **D**eep **N**onnegative **M**atrix **F**actorization (**CDNMF**). Firstly, we deepen NMF to strengthen its capacity for information extraction. Subsequently, inspired by contrastive learning, our algorithm creatively constructs network topology and node attributes as two contrasting views. Furthermore, we utilize a debiased negative sampling layer and learn node similarity at the community level, thereby enhancing the suitability of our model for community detection. We conduct experiments on three public real graph datasets and the proposed model has achieved better results than state-of-the-art methods. Code available at `https://github.com/6lyc/CDNMF.git`.

Index Terms— Community Detection, Deep NMF, Contrastive Learning, Community-Level Structure.
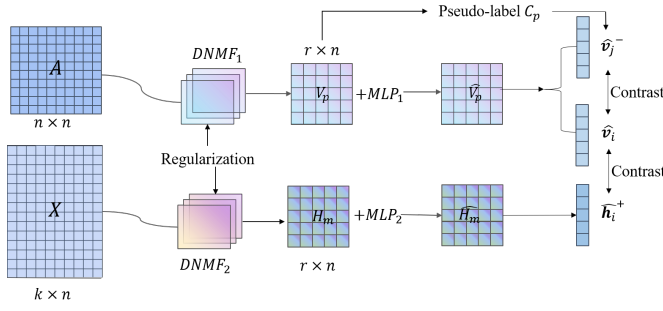
1. INTRODUCTION

Community detection (CD) is a fundamental task in complex network analysis. It involves partitioning a network into multiple substructures, each corresponding to a community. An effective partition requires nodes within the same community to be densely connected, while connections between nodes in different communities are sparse [1]. Mining the community structure is the key to revealing and understanding the

organizational principles and operation of complex network systems. For example, in social networks, platforms detect different user communities to facilitate friend recommendation and advertisement placement [2]. In addition to applications in social networks, CD is also widely used in protein-protein interaction (PPI) networks [3], citation networks [4], and more. Moreover, CD plays a crucial role in speaker diarization [5, 6].

Over the past two decades, many classical algorithms have been proposed for CD, such as modularity [7], conductance [8], and permanence [9]. However, these methods are only capable of assigning each node to one community, thereby narrowing their scope of applicability [10]. In recent years, certain neural network-based approaches have also been developed, including GUCD [11] and VGAER [12]. Nevertheless, their training processes resemble black boxes, leading to limited interpretability of the results. In addition, researchers have proposed community detection algorithms based on NMF [13], which have been widely used because of their good mathematical interpretability and natural applicability to the detection of overlapping communities. The NMF algorithm factorizes the adjacency matrix $A \in \mathbb{R}_+^{n \times n}$ of a graph in the following form: $A \approx UV$ s.t. $U \geq \mathbf{0}, V \geq \mathbf{0}$. Two nonnegative matrix factors $U \in \mathbb{R}_+^{n \times r}$ and $V \in \mathbb{R}_+^{r \times n}$ can be obtained. Pre-define the number of communities in the graph is r, the factor matrix U can be regarded as a mapping between the original network and the community membership space, with each column $u_i(i = 1, 2, \cdots, r)$ of U representing the basis vector of this space. The factor matrix V can be regarded as the node representation matrix (community indication matrix), with its element $v_{i,j}$ quantifying the tendency of the j-th node to belong to the i-th community. Overall, the factorization process and results of the NMF algorithm demonstrate strong interpretability. Additionally, when performing overlapping CD, naturally, we can also assign each node to several communities with relatively high tendency based on the matrix V. There are also some variants of NMF with better performance, such as Orthogonal Nonnegative Matrix Factorization (ONMF) [14], Bayesian Nonnegative Matrix Factorization (BNMF) [15], Nonnegative Matrix Tri-factorization (NMTF) [16] and MX-ONMTF [17] etc.

Fig. 1: The general framework of our Contrastive Deep Nonnegative Matrix Factorization (CDNMF).

However, these methods still suffer from some of the following three problems:

1. Such methods are shallow, often with only a single or double-layer mapping between the original network and the community membership space. Real-world networks have complex organizational principles, which are difficult to extract by shallow NMF.

2. Such methods tend to focus only on the topology of the network while ignoring the attributes of the nodes. In social sciences, it has been shown that the node attributes can reflect and influence the structure of their communities [18].

3. Such methods could not capture the community-level information in the network. NMF-based methods focus on the similarity between neighboring nodes and hardly extract global structural information necessary for CD.

This paper presents a new CD algorithm, called **CDNMF**, to address the limitations of existing methods. Overall, our main contributions are:

1. We propose to use Deep NMF (DNMF) as the backbone to enhance the representation learning capability of our model.

2. We construct a contrastive learning framework to unify the learning of graph topology and node attributes based on DNMF. Moreover, with the debiased negative sampling layer filtering out false negatives, our model better learns community-level information.

3. We conduct extensive experiments to evaluate the superiority, effectiveness, and efficiency of our CDNMF. The results show that our algorithm outperforms other SOTA CD methods.

2. PROPOSED METHOD

In this section, we describe our CDNMF in three modules. The general framework of the model is shown in Fig. 1.

2.1. DNMF Layer

In order to extract the hierarchical information from the original network topology, we reconstruct the adjacency matrix A as the deep form:

$$\min_{U_i, V_p} L_D = \|A - U_1 U_2 \ldots U_p V_p\|_F^2$$
$$\text{s.t. } V_p \geq \mathbf{0}, U_i \geq \mathbf{0}, \forall i = 1, 2, \ldots, p, \tag{1}$$

where $V_p \in \mathbb{R}_+^{r \times n}, U_i \in \mathbb{R}_+^{r_{i-1} \times r_i} (i = 1, 2, \ldots, p)$ and $n = r_0 \geq r_1 \geq \cdots \geq r_{p-1} \geq r_p = r$. r is the number of communities in the network. The $\|\cdot\|_F$ denotes the Frobenius norm of the matrix. Each matrix $U_i (i = 1, 2, \ldots, p)$ can be interpreted as the i-th feature matrix with different hierarchical information. $U_1 U_2 \ldots U_n$ is the total mapping between the original network and the community membership space. The matrix V_p is the node representation matrix after deep transformation, and each column can be understood as the tendency of a node to belong to different communities.

To transform the optimization problem with constraints in Eq. (1) into an unconstrained problem, we design a penalty term for each nonnegative matrix. At first, we define the function $f : \mathbb{R}^{a \times b} \to \mathbb{R}^{a \times b}$ for any matrix $B \in \mathbb{R}^{a \times b}$ satisfying:

$$f(B) = \begin{cases} B_{ij}, B_{ij} < 0 \\ 0, B_{ij} \geq 0. \end{cases} \tag{2}$$

In brief, the function f serves to convert the positive elements of the input matrix into 0 while leaving the negative elements unchanged. Naturally, the penalty term for the matrix $U_i (i = 1, 2, \ldots, p)$ is defined as:

$$\min_{U_i} \|f(U_i)\|_F^2. \tag{3}$$

Similarly, the penalty term for the matrix V_p is defined as:

$$\min_{V_p} \|f(V_p)\|_F^2. \tag{4}$$

Combining Eq. (3) as well as (4), we transform the optimization problem Eq. (1) into the following unconstrained objective function:

$$\min_{U_i, V_p} L_A = \|A - U_1 U_2 \ldots U_p V_p\|_F^2$$
$$+ \alpha \left(\sum_{i=1}^p \|f(U_i)\|_F^2 + \|f(V_p)\|_F^2 \right), \tag{5}$$

where $\alpha > 0$ is the coefficient of the nonnegative penalty term. Similarly, for the node feature matrix X of the network we obtain the following objective function:

$$\min_{W_j, H_m} L_X = \|X - W_1 W_2 \ldots W_m H_m\|_F^2$$
$$+ \alpha \left(\sum_{j=1}^m \|f(W_j)\|_F^2 + \|f(H_m)\|_F^2 \right). \tag{6}$$

6726

Overall, we obtain the objective function for the DNMF term:

$$\min_{U_i, V_p, W_j, H_m} L_{DNMF} = L_A + L_X. \tag{7}$$

In addition, to preserve the intrinsic geometric structure of node pairs in deep hierarchical mapping, we introduce graph regularization as follows [16]:

$$\frac{1}{2} \sum_i \sum_j A(i,j) \|V_p(:,i) - V_p(:,j)\|_2^2 = \mathrm{tr}\left(V_p L V_p^T\right), \tag{8}$$

where $L = D - A$ denotes the graph Laplacian matrix, D is a diagonal matrix, and its diagonal elements are the row sums of A. $\mathrm{tr}(\cdot)$ denotes the trace of the matrix.

Then, we minimize the graph regularization for two views and obtain the objective function for the graph regularization term:

$$\min_{V_p, H_m} L_{reg} = L_{reg_A} + L_{reg_X}$$
$$= \mathrm{tr}\left(V_p L V_p^T\right) + \mathrm{tr}\left(H_m L H_m^T\right). \tag{9}$$

2.2. Debiased Negative Sampling Layer

After the DNMF layer, we would obtain the pseudo community labels from the node representation matrix, which could reduce the false negative samples. Specifically, we first obtain the pseudo labels of node v_i by:

$$c_i^* = \mathrm{argmax}\left(V_p(:,i)\right) \tag{10}$$

Next, we remove all nodes that have the same pseudo label as v_i to obtain the debiased negative sample set $\widetilde{\mathcal{N}}_i$ of node v_i:

$$\widetilde{\mathcal{N}}_i = \{v_m\}\left(c_m^* \neq c_i^*\right). \tag{11}$$

In fact, the accuracy of pseudo community labels continues to improve with training iterations.

2.3. Graph Contrastive Learning Layer

As shown in Fig. 1, we use the adjacency matrix A of the network topology and the feature matrix X of the node attributes as two views for contrastive learning. In particular, for each node v_i, we consider the representation vector $V_p(:,i)$ generated on the topology view as the anchor point, the representation vector $H_m(:,i)$ generated on the node attribute view as the positive sample, and the representation vectors of nodes in the set $\widetilde{\mathcal{N}}_i$ in Eq. (11) as the negative samples.

Then, for each positive sample pair $(V_p(:,i), H_m(:,i))$, we define the contrastive loss as follows:

$$l\left(V_p(:,i), H_m(:,i)\right)$$
$$= \log \frac{e^{\theta(V_p(:,i), H_m(:,i))/\tau}}{e^{\theta(V_p(:,i), H_m(:,i))/\tau} + \sum_{k=1}^n \mathbb{1}_{\left[k \in \widetilde{\mathcal{N}}_i\right]} e^{\theta(V_p(:,i), V_p(:,k))/\tau}},$$
$$\tag{12}$$

Algorithm 1 CDNMF

Input: a graph $G = (A, X)$, the number of communities r;
Output: C_p;
1: **Pre-training stage**:
2: $U_1, V_1 \leftarrow \mathrm{NMF}\left(A, r_1\right)$;
3: **for** $i = 2$ to p **do**
4:    $U_i, V_i \leftarrow \mathrm{NMF}\left(V_{i-1}, r_i\right)$; $//r_p = r$
5: **end for**
6: The same pre-training process for X.
7: **Fine-tuning stage**:
8: Initialize each matrix factor with the pre-training values.
9: **for** $i = 1$ to $epoch$ **do**
10:    Generate the pseudo labels c_i^* by Eq. (10).
11:    Update the negative sample set $\widetilde{\mathcal{N}}_i$ by Eq. (11).
12:    Update model parameters by minimizing Eq.(14) through SGD.
13: **end for**
14: **return** C_p

---

where $\theta(\boldsymbol{v}, \boldsymbol{h}) = s(g(\boldsymbol{v}), g(\boldsymbol{h}))$, s is the cosine similarity function, and g is the MLP of two layers. $\mathbb{1}_{\left[k \in \widetilde{\mathcal{N}}_i\right]} \in \{0, 1\}$ is the indicator function that equals 1 if $k \in \widetilde{\mathcal{N}}_i$ and 0 otherwise. τ is the temperature parameter.

The contrast of positive samples extracts consistency as well as complementary information from each node topology and attribute. The contrast of negative samples expands the distance between different communities so that our model learns the community-level similarity between nodes.

In the graph contrastive learning layer, we optimize the contrastive loss of each node, and obtain the objective function for the contrastive learning term:

$$\min_{V_p, H_m} L_{cl} = -\frac{1}{n} \sum_{i=1}^n l\left(V_p(:,i), H_m(:,i)\right) \tag{13}$$

2.4. Training Process

We jointly optimize each layer, and define the total objective function as follows:

$$\min_{U_i, V_p, W_j, H_m} L = L_{DNMF} + \beta L_{reg} + \gamma L_{cl}, \tag{14}$$

where $\beta, \gamma > 0$ are the scale factors. After the training process in Algorithm 1, the predicted community labels for each node v_i are:

$$C_p = \{c_i^*\}_{i=1}^n = \mathrm{argmax}\left(V_p\right). \tag{15}$$

3. EXPERIMENTS

In this section, we first introduce our experimental setup and then compare our method with state-of-the-art methods in community detection tasks.

Table 1: Community detection performance with ACC and NMI on three datasets. The **bold** and underlined text indicate the optimal and suboptimal results, respectively.

Method	Cora		Citeseer		PubMed	
	ACC	NMI	ACC	NMI	ACC	NMI
NMF	0.4103	0.2851	0.3074	0.1319	0.5133	0.1606
ONMF	0.3811	0.2416	0.3330	0.1423	0.5575	0.1582
BNMF	0.4191	0.2521	0.3324	0.0825	0.5110	0.0714
NSED	0.4234	0.2928	0.3448	0.1492	0.5201	0.1729
LINE	0.4044	0.2376	0.3019	0.0573	0.4990	0.1357
Node2Vec	0.3674	0.1978	0.2521	0.0486	0.4067	0.0635
MNMF	0.1647	0.0035	0.1890	0.0031	0.3397	0.0002
LP-FNMTF	0.2861	0.0261	0.2327	0.0143	0.5437	0.1532
K-means++	0.3230	0.2210	0.4160	0.1910	0.4150	0.2300
VGAER	0.4530	0.2970	0.3020	0.2170	0.3010	0.2230
DNMF	0.4849	0.3572	0.3635	0.1582	0.5389	0.1709
DANMF	0.5499	0.3764	0.4242	0.1831	0.6393	0.2221
Ours	**0.6081**	**0.4006**	**0.4756**	**0.2559**	**0.6653**	**0.2330**

Table 2: Results of ablation experiments based on Cora and Citeseer.

Methods	Cora				Citeseer			
	ACC	Δ	NMI	Δ	ACC	Δ	NMI	Δ
Ours $[L(A)]$	0.5835	2.46%	0.3781	2.25%	0.4598	1.58%	0.1672	8.87%
Ours $[L(X)]$	0.5162	9.19%	0.3501	5.05%	0.3499	12.6%	0.1749	8.10%
Ours	**0.6081**		**0.4006**		**0.4756**		**0.2559**	

Fig. 2: The analysis of the convergence rate of our algorithm.

3.1. Experimental Setup

We perform experiments on three widely used graph datasets: Cora [19], Citeseer [20], and PubMed [21]. Then, we compare our method with four types of CD methods, including four shallow NMF-based methods: NMF [22], ONMF [14], BNMF [15], NSED [23], three network embedding methods: LINE [24], Node2Vec [25], MNMF [26], three methods that consider node attributes: LP-FNMTF [27], K-means++ [28], VGAER [12] and two methods based on Deep NMF: DNMF [10], DANMF [10].
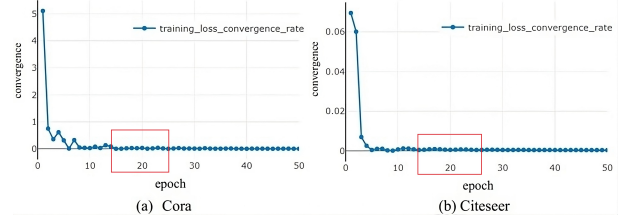
In particular, we fixed the number of hidden layers in the model to 3. The parameters used for the Cora are: $\alpha = 400, \beta = 3.0, \gamma = 5.0, \tau = 1.3$; the parameters used for the Citeseer are: $\alpha = 3000, \beta = 1.0, \gamma = 5.0, \tau = 1.5$; and the parameters used for the PubMed are: $\alpha = 100, \beta = 1.0, \gamma = 1.0, \tau = 0.5$. Moreover, ACC and NMI will be used for evaluating each CD method. We run each algorithm 20 times and report the average results.

3.2. Community Detection

It is worth noting that DNMF steadily obtains better performance than shallow NMF-based methods since it can learn hierarchical information. Our CDNMF obtains better performance than DNMF because it learns consistent semantic information from both network topology and node features. Moreover, our model expands the distance between different communities to fully learn the community-level similarity between nodes. Although all three embedding-based approaches attempt to maintain higher-order similarity between nodes, they do not exhibit competitive performance. For LINE and Node2Vec, they do not expressly model to preserve community-level similarity between nodes. In contrast, MNMF applies modularity to reveal the community structure of the network. However, the modularity approach may suffer from the Resolution Limit Problem [29]. The results are shown in Table 1.

3.3. Ablation Experiments

We consider removing our model's graph contrastive learning layer, i.e., let $\gamma = 0$ in the total objective function. Then we perform community detection experiments based on the adjacency matrix A and the node feature matrix X, respectively, denoted as "Ours $[L(A)]$" and "Ours $[L(X)]$".

In both datasets, "Ours $[L(A)]$" basically outperforms "Ours $[L(X)]$", which indicates that the topology of the network has more influence on community detection to some extent. However, both of them are less effective than "Ours". This indicates that the information from a single view, either the topology of the network or the attributes of the nodes, cannot accurately model the community-level relationships between nodes. The results are shown in Table 2.

3.4. Convergence Analysis

Our optimization process is divided into two stages. In the pre-training stage, we perform NMF for each layer of the two input matrices, and its convergence has been analyzed in [23]. Next, we visualize the convergence of the fine-tuning stage in Fig. 2. We observe that our model can achieve convergence within about 20 epochs with a fast convergence rate. Similar results also can be observed on PubMed.

4. CONCLUSION

In this paper, we propose a novel community detection method called CDNMF, which innovatively combines the idea of graph contrastive learning into NMF. Our CDNMF extracts the consistency information in network topology and node features through positive sample pairs and expands the distance of different communities in the representation space through negative sample pairs. We conduct various experiments to verify the superiority, effectiveness, and efficiency of our model. Moreover, there is potential to integrate additional matrix factorization and contrastive learning algorithms.

5. REFERENCES

[1] Michelle Girvan and Mark EJ Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[2] Jierui Xie and Boleslaw K Szymanski, "Towards linear time overlapping community detection in social networks," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2012, pp. 25–36.

[3] Ziqi Gao, Chenran Jiang, Jiawen Zhang, Xiaosen Jiang, Lanqing Li, Peilin Zhao, Huanming Yang, Yong Huang, and Jia Li, "Hierarchical graph learning for protein–protein interaction," *Nature Communications*, vol. 14, no. 1, pp. 1093, 2023.

[4] Fanzhen Liu, Shan Xue, Jia Wu, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Jian Yang, and Philip S Yu, "Deep learning for community detection: Progress, challenges and opportunities," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, July 2020, IJCAI-PRICAI-2020.

[5] Jie Wang, Zhicong Chen, Haodong Zhou, Lin Li, and Qingyang Hong, "Community detection graph convolutional network for overlap-aware speaker diarization," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[6] Siqi Zheng and Hongbin Suo, "Reformulating speaker diarization as community detection with emphasis on topological structure," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8097–8101.

[7] Mark EJ Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[8] Jure Leskovec, Kevin J Lang, and Michael Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 631–640.

[9] Tanmoy Chakraborty, Sriram Srinivasan, Niloy Ganguly, Animesh Mukherjee, and Sanjukta Bhowmick, "On the permanence of vertices in network communities," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 1396–1405.

[10] Fanghua Ye, Chuan Chen, and Zibin Zheng, "Deep autoencoder-like nonnegative matrix factorization for community detection," in *Proceedings of the 27th ACM international conference on information and knowledge management*, 2018, pp. 1393–1402.

[11] Dongxiao He, Yue Song, Di Jin, Zhiyong Feng, Binbin Zhang, Zhizhi Yu, and Weixiong Zhang, "Community-centric graph convolutional network for unsupervised community detection," in *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, 2021, pp. 3515–3521.

[12] Chenyang Qiu, Zhaoci Huang, Wenzhe Xu, and Huijia Li, "Vgaer: graph neural network reconstruction based community detection," *arXiv preprint arXiv:2201.04066*, 2022.

[13] Chaobo He, Xiang Fei, Qiwei Cheng, Hanchao Li, Zeng Hu, and Yong Tang, "A survey of community detection in complex networks using nonnegative matrix factorization," *IEEE Transactions on Computational Social Systems*, 2021.

[14] Filippo Pompili, Nicolas Gillis, P-A Absil, and François Glineur, "Two algorithms for orthogonal nonnegative matrix factorization with application to clustering," *Neurocomputing*, vol. 141, pp. 15–25, 2014.

[15] Ioannis Psorakis, Stephen Roberts, Mark Ebden, and Ben Sheldon, "Overlapping community detection using bayesian non-negative matrix factorization," *Physical Review E*, vol. 83, no. 6, pp. 066114, 2011.

[16] Yulong Pei, Nilanjan Chakraborty, and Katia Sycara, "Nonnegative matrix tri-factorization with graph regularization for community detection in social networks," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[17] Meiby Ortiz-Bouza and Selin Aviyente, "Orthogonal nonnegative matrix tri-factorization for community detection in multiplex networks," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 5987–5991.

[18] Miller McPherson, Lynn Smith-Lovin, and James M Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, pp. 415–444, 2001.

[19] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad, "Collective classification in network data," *AI magazine*, vol. 29, no. 3, pp. 93–93, 2008.

[20] Ryan Rossi and Nesreen Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proceedings of the AAAI conference on artificial intelligence*, 2015, vol. 29.

[21] Galileo Namata, Ben London, Lise Getoor, Bert Huang, and U Edu, "Query-driven active surveying for collective classification," in *10th international workshop on mining and learning with graphs*, 2012, vol. 8, p. 1.

[22] Shawn Mankad and George Michailidis, "Structural and functional discovery in dynamic networks with non-negative matrix factorization," *Physical Review E*, vol. 88, no. 4, pp. 042812, 2013.

[23] Bing-Jie Sun, Huawei Shen, Jinhua Gao, Wentao Ouyang, and Xueqi Cheng, "A non-negative symmetric encoder-decoder approach for community detection," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 597–606.

[24] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

[25] Aditya Grover and Jure Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[26] Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiqiang Yang, "Community preserving network embedding," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[27] Hua Wang, Feiping Nie, Heng Huang, and Fillia Makedon, "Fast nonnegative matrix tri-factorization for large-scale data co-clustering," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[28] Sergei Vassilvitskii and David Arthur, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2006, pp. 1027–1035.

[29] Santo Fortunato and Marc Barthélemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences of the United States of America*, pp. 36–41, 2007.