Deep learning model for community detection fusing network structure and node attributes

Huirong Wang, Peng Liu, Liang Gui*
School of Economics and Management, Jiangsu University of Science and Technology, Zhenjiang 212003, Jiangsu, China

ABSTRACT

Community detection is a research hotspot in network science. Most of the existing discovery methods use edges to represent the similarity of attributes between nodes to implement community exploration. However, empirical studies have shown that there are many other factors besides the similarity of node attributes (e.g., heterophily). In this work, a community discovery model (VNSA) is proposed based on variational graph autoencoders that can fuse network structure and node attributes. Experimental results based on real networks show that this model can effectively complete community detection tasks, and its performance is significantly improved compared with traditional methods (such as Louvain) and deep learning methods (such as Deepwalk). The Model not only better reflects the idea of "homogeneity attracts" in community division but also has certain reference value for relevant practical applications such as friendship recommendations.

Keywords: Community detection, community discovery model, VNSA, deep learning

1. INTRODUCTION

Complex networks are often used to describe real systems in various domains, such as social systems, biological systems, and technological systems. These systems always demonstrate a community (modular) structure from the perspective of the complex network[1]. Research on community detection in networks can be divided into two phases.

The first stage of the work is the traditional community discovery algorithm, mainly through the traditional statistical inference and machine learning methods. For example, the community detection algorithm based on modularity optimization widely used today (such as the GN algorithm[2], FN algorithm[3], Louvain algorithm[4], etc.) mainly adopts the idea of hierarchical clustering which repeatedly adjusts the community according to the number of connections, so as to achieve the optimal modularity and realize the community division of the network. However, with the deeper research on community detection, traditional community detection algorithms are difficult to face complex system scenarios, especially due to the fact that algorithms are highly dependent on data characteristics[5].

With the generation and propulsion of deep learning research, the research work of community detection in the network has entered the second stage. Not only can the deep learning model learn the representation of network data at multiple levels of abstraction, but also it has a strong learning ability of nonlinear characteristics[6]. What's more, the deep neural network model can greatly enrich the community exploration tasks in the network. Currently, community discovery methods in the field of deep learning mainly use deep neural networks, deep graph embedding, and graph neural network models.

The first model can better capture and model complex relationships, including convolutional neural networks (CNN, convolutional neural networks), autoencoders (AE, auto-encoders), and generative adversarial networks (GAN, generative adversarial networks). For example, Xin *et al.* proposed a new CNN model to detect communities in networks with missing part of edge information[7]. Jang proposed a convolutional variational autoencoder (CVAE)[8]. In order to make use of the attribute characteristics of nodes, Cao *et al.* proposed a cascading autoencoder that integrates network structure and node attributes to achieve community division[9]. The community discovery method based on the depth map embedding model is to map node information to lower-dimensional vector space while preserving structural information as much as possible and then implement community division by machine learning method. For example, Xie *et al.* proposed a network representation method based on deep sparse filtering[10], which is an unsupervised network feature

* jace.gui@qq.com

Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)
edited by Yingfa Lu, Changbo Cheng, Proc. of SPIE Vol. 12506, 1250664
 · 0277-786X · doi: 10.1117/12.2661768

extraction method, especially suitable for community discovery of large-scale networks. The community discovery method based on a graph neural network is essentially the fusion of graph mining technology and deep learning, the core of which is to form aggregate features of node structure information through deep learning technology and then realize the community division of the network. For example, Chen *et al.* defined the adjacency matrix by using a non-backtracking operator, and on this basis of which, it completed the community division task through supervised learning[11].

Based on the brief review of relevant work above, it is not difficult to learn that the deep learning model has further promoted the development of community detection research on complex networks, but the work in this area is still in the exploratory stage which deserves further development. On the one hand, the core idea of community division in complex networks stems from the theory of "homogeneity attracts" in sociology, and the connection between nodes is regarded as a reflection of the interaction between individuals with similar attributes in the system[12]. However, under the theory of "difference preference" in sociology, individuals with different attributes in the system can also establish connections. Therefore, it is inevitable to deviate from the core idea of "homogeneity attracts" to divide communities by reflecting the attributes and characteristics of individuals who establish links and exploring network communities by only using linking relations. On the other hand, the existing network community detection methods using a deep learning model often adopt supervised or semi-supervised learning, besides, the number of communities needs to be specified in advance, hence it is difficult to apply to the situation where the number of communities is unknown. Therefore, the unsupervised deep learning method that integrates the edge information and attribute information of nodes can not only better reflect the core idea of community division that "homogeneity attracts", but also apply more effectively the community division method to realistic scenes such as system recommendation[13] and social opinion mining[14]. At the same time, some studies have found that excessive reliance on the structural information of nodes in recommendation tends to make the development of network structure appear to be differentiated, and the addition of node attributes can effectively alleviate the differentiation of the network[15]. This means that community division only based on network structure information may hurt the integrity and stability of the subsequent development of the network when community structure is used for system recommendation. Therefore, it is necessary to take the attribute information of nodes into consideration in community detection.

In this regard, this paper proposes a deep learning model that integrates network structure and node attributes. The model extracts the attributes of network nodes, and by adding convolutional neural networks and community division indicators, it can achieve effective convergence between similarity nodes, better reflect the idea of "homogeneity attracts" in community division, and also provide new ideas for real community division tasks such as community opinion mining.

2. VNSA MODEL

According to the research work of Kipf and Welling[16], VGAE is to encode the known graph to learn the distribution of its node vector representation, then sample the node vector representation from the distribution, and finally decode to reconstruct the graph structure. Hence, the VNSA model proposed in this paper draws on the encoding process of the VGAE, integrates node attributes, and introduces the community division index. The framework of the model mainly includes three steps, as shown in Figure 1.

(1) Node feature extraction

The real network not only has a certain topological structure, but also the attribute characteristics of its nodes. Therefore, the first step of the model proposed in this paper is to extract the attributes and edge relations of nodes in the network. Specifically, for any network containing n nodes, each node has m-dimensional attributes, and the attribute vector of each node is combined into the attribute matrix $M_T^{n\times m}$.The adjacency matrix $M_A^{n\times n}$ of the network is obtained from the connection relation between nodes, and the matrix is $\tilde{M}_A$ obtained by Laplace normalization.

(2) Community division

Community division mainly draws on the part of VGAE coding which adopts the multi-layer convolutional neural network to achieve the embedding of the network. Meanwhile, some studies have shown that when the number of convolutional layers exceeds two, the learning effect is not significantly improved. Therefore, on the basis of (1), a two-layer convolutional network is selected to achieve network embedding, as shown in equations (1) and (2) respectively. In

the equation (1), $Z_{(1)}^{n\times p}$ represents the node characteristic matrix obtained after the first convolution layer operation, $W$is the hyperparameter matrix of $m \times p$ $(q < p < m)$, where the parameter q is the given number of divided communities. In equation (2), $Z_{(2)}^{n\times q}$ is the sign matrix of nodes obtained through the second convolution layer.

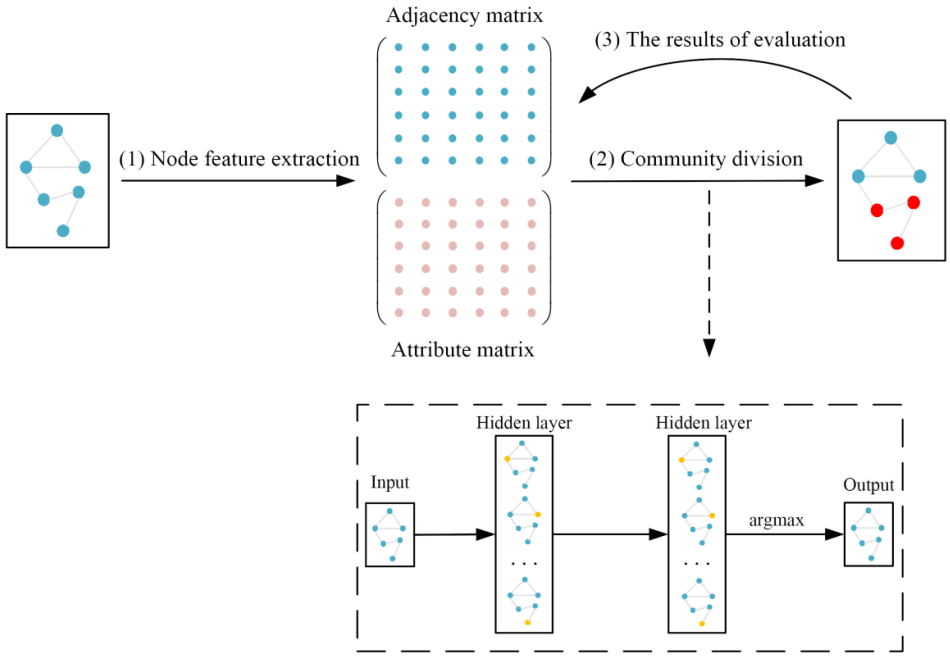

Figure 1. Framework of VNSA model.

$$Z_{(1)}^{n\times p} = \tilde{M}_A M_T^{n\times m} W_0 \tag{1}$$

$$Z_{(2)}^{n\times q} = \tilde{M}_A Z_{(1)}^{n\times p} W_1 \tag{2}$$

According to the network embedding result $Z_{(2)}^{n\times q}$, this paper uses the argmax activation function to determine the most likely community to which the node belongs, and then obtain the community division result.

(3) The results of evaluation

According to the goal of community division, the results of community division are evaluated from two aspects: modularity (Q) and average similarity degree of node attributes within the community (S), and the result score E is recorded as shown in equation (3). Q value adopts the traditional module-degree calculation method, S value is the Euclidean distance after the normalization of node attribute vector, and w is the community division target weight, of which the value ranges from 0 to 1. Then, within a given range, adjust the value of parameter q, repeat steps (2), and finally select the community division result with the maximum value of E. According to Equation (3), when the value of w is small, if E wants to reach the maximum value, the value of S learned by the model can only be increased, thereby indirectly learning the mechanism of heterogeneity. Instead, the model can learn the mechanism of homogeneity.

$$E = w\times Q + (1-w)\times S \tag{3}$$

3. EMPIRICAL ANALYSIS

This section takes two knowledge collaboration networks in two different fields as examples, one in the field of technology research, and another in the field of open-source software development. Then, this section uses the proposed

model to divide them into communities to test the effectiveness of the model.

3.1 Data set and network construction

In the technology research and development field, this paper collects a total of 5000 patent data under the subcategory "Digital Information Transmission" (IPC classification code H04L) from the incoPat database. Furthermore, the inventor in each patent data is abstracted as a node, and the cooperative relationship between co-inventors is abstracted as an edge, so as to build an inventor collaboration network (ICN). The attribute characteristics of each inventor are described by the vector of the number of patents granted by each inventor in fourteen large groups of subclasses under H04L.

In the field of open-source software development, this paper selects the developer community of the current mainstream front-end software Angular and collects 250,423 submission records from this community from June 2013 to August 2019. Each submission record contains the developer's email address, the type of submission, and the files involved in the code changes. Since open-source software is distributed in the release form, each corresponding software release can be viewed as the collaborative knowledge of all developers within the release cycle. Therefore, this paper abstracts the email addresses in the submission records as nodes and regards the email addresses that have submitted codes to the same files in the same software version development cycle as cooperative relations and abstracts them as edges to build a developer collaboration network (DCN). On this basis, different types of submission records for each developer are counted and vectors are constructed as developer attributes. The basic information of the two networks is shown in Table 1.

Table 1. Basic information of the instance networks.

Network	node	edge	Node attribute dimensions
ICN	923	2069	14
DCN	1439	5166	17

3.2 Results analysis

For the constructed network, the method proposed in this paper is used to divide the community, and relevant parameter settings are shown in Table 2.

Table 2. Parameter setting.

Parameter	Numerical setting	
	ICN	DCN
Convolution layer	2	2
Training times	100	100
Learning rate	0.01	0.01
Input layer dimensions	14	17
Range of hidden layer dimensions	[9, 7]	[13, 11]
Range of output layer dimensions	[8, 6]	[11, 9]
Dividing target weight (w)	0.3	0.3

In Table 2, considering that community division of a complex network is essentially a clustering process of nodes, the number of communities divided must be lower than the number of node attribute dimensions. Therefore, the maximum variation range of hidden dimension numbers should not exceed the number of node dimensions. In addition, in order to clearly observe whether the method proposed in this paper effectively achieves the convergence of similar nodes in community division, the weight value (w) of the community division objective was set at 0.3 in the experiment. Under the combination of the above two networks with different hidden and output layer dimensions, the score changes of community division results are shown in Table 3.

Table 3. Results under different dimension numbers of the hidden layer (hl) and output layer (ol).

(hl,ol)	ICN			(hl,ol)	DCN		
	Q	***S***	***E***		***Q***	***S***	***E***
(11,8)	0.33	0.23	0.26	(13,9)	0.23	0.28	0.26
(11,7)	0.32	0.25	0.27	(13,8)	0.35	0.29	0.31
(11,6)	0.36	0.30	0.32	(13,7)	0.37	0.31	0.33
(10,8)	0.38	0.31	0.33	(12,9)	0.42	0.33	0.36
(10,7)	0.42	0.33	0.36	(12,8)	0.41	0.33	0.35
(10,6)	0.39	0.32	0.34	(12,7)	0.39	0.32	0.34
(9,8)	0.41	0.27	0.31	(11,9)	0.32	0.31	0.31
(9,7)	0.38	0.28	0.31	(11,8)	0.30	0.29	0.30
(9,6)	0.36	0.23	0.27	(11,7)	0.31	0.23	0.26

It can be seen from Table 3 that in the DCN when the number of hidden layer dimensions is 13, node similarity (*S*) and partition result score (*E*) in the community gradually increase with the decrease of the number of output layer dimensions. When the number of hidden layer dimensions drops to 12, the values of the above indicators decrease with the decrease of the number of output layer dimensions. Therefore, when the number of hidden layers is 12 and the number of output layers is 9, *E* reaches the maximum value, which represents the best community division result of DCN. At this point, the DCN can be divided into nine communities. Similarly, the ICN can be divided into seven communities.

3.3 Model comparative analysis

To further verify the effectiveness of the proposed model, we compare the model with the VNSA model without fusing node attributes (Non-VNSA), the Louvain and Spectral Clustering method in the traditional method, and the Deepwalk and Node2vec in the deep learning method. Specific results are shown in Table 4 below.

Table 4. Comparison of community detection under different methods.

Methods	ICN			DCN		
	Q	***S***	***E***	***Q***	***S***	***E***
Louvain	0.42	0.11	0.20	0.44	0.11	0.21
Spectral Clustering	0.31	0.08	0.15	0.37	0.13	0.20
Deepwalk	0.22	0.18	0.20	0.18	0.19	0.18
Node2vec	0.28	0.10	0.16	0.24	0.13	0.16
Non-VNSA	0.42	0.11	0.20	0.43	0.11	0.21
VNSA	0.43	0.32	0.35	0.42	0.33	0.36

Table 4 shows that the model achieves the highest community division score. In the DCN, although Louvain obtained the highest modularity (Q=0.44), the other indexes of the model were much higher than it and the modularity Q was close to it. Thus the VNSA model can better realize the convergence between nodes with similar attributes while ensuring that the network is highly modular.

In general, the model can be effectively applied in the field of community detection. The results show that the modularity is not lower than traditional methods (such as the Louvain method) and deep learning methods (such as the Node2vec method), but the similarity of nodes is significantly higher. This demonstrates that the model can better reflect the core idea of "homogeneity attracts" when completing the tasks of community division, and also lay a foundation for the performance improvement of practical applications based on community division (such as system recommendation).

4. CONCLUSION

In this paper, a community discovery model (VNSA) for complex networks is proposed, which takes both node structure and attribute information into account. Compared with some traditional methods (Louvain and Spectral Clustering) and deep learning methods (Node2vec and Deepwalk), the proposed model is validated. In terms of node similarity comparison, the model can better reflect the core idea of "homogeneity attracts", and provide some exploration for network community exploration tasks (e.g., e-commerce recommendation) that need to be considered more node attribute information. In the following work, the author intends to further test the effectiveness of the proposed method through larger-scale network data, study the optimization of the fusion model of node connection information and attribute information in the method, so as to apply it to more complex network community detection scenarios (such as community detection of heterogeneous networks).

REFERENCES

[1] Spirin, V. and Mirny, L. A., P. Natl. Acad. Sci., 100, 12123-28(2003).
[2] Girvan, M. and Newman, M. E. J., P. Natl. Acad. Sci., 99, 7821-26(2002).
[3] Newman, M. E. J., Phys. Rev. E., 69, 066133(2004).
[4] Blondel, V. D., Guillaume, J. L. and Lambiotte, R., J. Stat. Mech. Theory Exp., 2008, P10008(2008).
[5] Fortunato, S. and Barthelemy, M., P. Natl. Acad. Sci., 104, 36-41(2007).
[6] Liu, F., Xue, S., Wu, J., Zhou, C., Hu, W., Paris C., Nepal, S. and Yu, P. S., IJCAI, 4981-87(2020).
[7] Xin, X., Wang, C., Ying, X. and Wang, B., Physica A., 469, 342-52(2017).
[8] Jang, J. H., Kim, T. Y., Lim, H. S. and Yoon, D. K., PLoS One.16(2021).
[9] Cao, J., Jin, D., Yang, L. and Dang, J., Neurocomputing, 297, 71-81(2018).
[10] Xie, Y., Gong, M., Wang, S. and Yu, B., Pattern Recognit., 81, 50-9(2018).
[11] Chen, Z., Li, L. and Bruna, J., [arXiv:2020.1705.08415[stat.ML]], (2020).
[12] Ferreyra, N. E. D., Hecking, T., Aïmeur, E., Heisel, M. and Hoppe, H. U., OSNEM, 29, 100203(2022).
[13] Li, Y., Han, Q. and Liu, J., J Phys Conf Ser., 1345, 032055(2019).
[14] Huang, X., Chen, D., Ren, T. and Wang, D., Data Min. Knowl. Discov., 35, 1-45(2021).
[15] Santos, F. P., Lelkes, Y. and Levin, S., P. Natl. Acad. Sci., 118, e2102141118(2021).
[16] Kipf, T. N. and Welling, M., [arXiv:2016.1611.07308[stat.ML]], (2016).