Community Detection in Social Networks using Deep Learning

Dhilber M [1] **and S Durga Bhavani** [1,*]

[1] *School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India*

Correspondence*:
S Durga Bhavani
sdbcs@uohyd.ernet.in

ABSTRACT

Community structure is found everywhere from simple networks to real world complex networks. Detecting communities in such networks is always a challenging problem in the area of network theory. The task of community detection has a wide variety of applications ranging from recommendation systems, advertising, marketing, epidemic spreading, cancer detection etc. The two mainly existing approaches for community detection, namely, stochastic model and modularity maximization model focus on building a low dimensional network embedding to reconstruct the original network structure. However the mapping to low dimensional space in these methods is purely linear. Understanding the fact that real world networks will have non-linear structure in abundance, aforementioned methods become less practical for real world networks. Considering the nonlinear representation power of deep neural networks, several solutions based on autoencoders are being proposed. In this work, we propose a new method wherein we stack multiple autoencoders and apply parameter sharing. This method of training autoencoders has been successfully applied for the problems of link prediction and node classification in the recent literature. Our enhanced model with modified architecture produced better results compared to many other existing methods. We tested our model on a few benchmark datasets and obtained competitive results.

Keywords: community detection, social networks, deep learning, modularity, autoencoders

1 INTRODUCTION

Detecting community structures in networks is a vital task in network theory. There have been a fair amount of approaches in achieving this task in the literature (E.J. Newman, 2006a; Psorakis et al., 2011). In this increasingly interlinked world, studying and analyzing relationships and patterns in networks is becoming inevitable.

Recently there has been a lot of work that incorporates non-linear capabilities of deep neural networks (E. Hinton and S. Zemel, 1994) in achieving community detection and related tasks (Yang et al., 2016; Vu Tran, 2018; Perozzi et al., 2014; Grover and Leskovec, 2016; Hamilton et al., 2017). Our work is closely related to (Yang et al., 2016), a new method of community detection in which network embeddings are used to detect communities. However we followed a different architecture and training scheme for the deep neural network which gave us better results.

2 COMMUNITY DETECTION PROBLEM

Solutions to the problem of community detection can be broadly classified into Modularity maximization model (E.J. Newman, 2006a) and Stochastic model (He et al., 2015; Jin et al., 2015; E. Hinton and S. Zemel, 1994). Modularity maximization model introduced by Newman in 2006 to maximize the modularity function Q, where modularity Q is defined as difference between number of edges within community and the expected number of edges over all pair of vertices.

$$Q = \frac{1}{4m}\sum_{i,j}(a_{ij} - \frac{k_i k_j}{2m})(h_i h_j)$$
$$= \frac{1}{4m}h^T Bh$$

Here h is the community membership vector with $h_i = 1$ or -1 to denote the two different communities, k_i denotes the degree of node i, B is the modularity matrix and m is the total number of edges in network.

In Stochastic model, community detection is formulated as Non Negative Matrix Factorization problem (NMF). This approach aims to find a non-negative membership matrix H to reconstruct adjacency matrix A. (Yang et al., 2016) prove that both Modularity maximization and Stochastic model can be interpreted as finding low dimensional representations to best reconstruct new structure. They further investigate that mapping of networks to lower dimensions is purely linear, which makes them less practical for real world networks and hence propose a non-linear solution using autoencoders (E. Hinton and S. Zemel, 1994).

3 RELATED WORK

Recently tremendous research work is reported on network embedding based approaches in solving community detection and related problems. (Perozzi et al., 2014) uses random walks to learn embedding by considering random walks as equivalent to sentences for language representation problems. An improved approach (Grover and Leskovec, 2016) implements a biased random walk which explores diverse neighbhorhoods. (Kipf and Welling, 2016) applies convolutional neural networks(CNN) directly on graph structured data to encode features. (Jia et al., 2019) makes use of Generative Adverserial Networks to obtain representations with membership strength of vertices in communities and solves overlapping community detection problem. (Vu Tran, 2018) build autoencoders with tied weights for multi task learning of link prediction and node classification. (Hamilton et al., 2017) introduces an inductive framework to generate node embedding for unseen networks by modelling a neighbourhood aggregation function. Our work is closely related to (Yang et al., 2016) which uses autoencoders to obtain latent representations. However, our method is different in both model architecture as well as training scheme which is explained in following sections.

4 OUR PROPOSED MODEL

(Yang et al., 2016) are among the first researchers to apply deep learning(DL) approach to the community detection problem. They construct a DL model by connecting autoencoders in series to obtain parameter optimization. They train the first autoencoder to reduce reconstruction error, take new representation obtained from the first autoencoder and give it to the next one and so on. The key differences between our model and (Yang et al., 2016) are, first, we stack the autoencoder layers together, not in series and perform common training to all the layers instead of training each autoencoder separately. Secondly, we apply

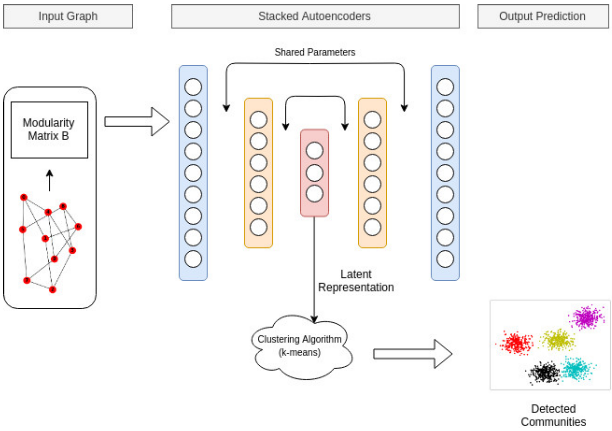

Figure 1. Architecture of the proposed model.

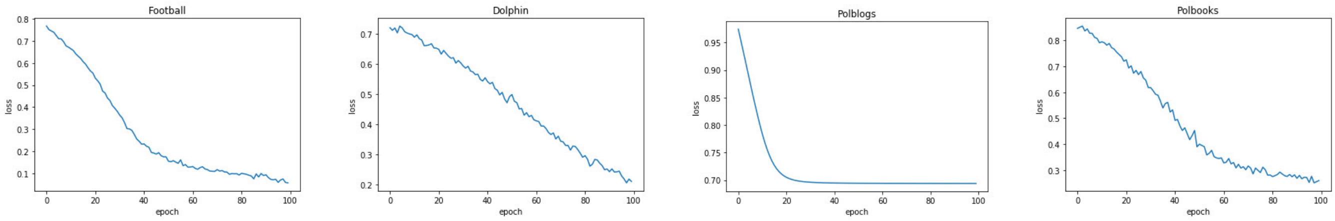

Figure 2. Comparison of loss with respect to number of epochs for different datasets

parameter sharing across layers to control the parameter growth and obtain optimization. This method of weight tying was successfully applied by (Vu Tran, 2018) in solving link prediction and node classification problems simultaneously.

4.1 Method

The method of (Yang et al., 2016) with minor changes is implemented which is discussed in this section. First an autoencoder model with a common 2 layer architecture is built with varied layer configurations depending on datasets. We train the autoencoder for multiple iterations to reduce the reconstruction error. The modularity matrix of each network is given as input to the autoencoder and a low dimensional representation of the corresponding network is obtained. Community detection is carried out by applying clustering algorithms (K-means in this case) to the obtained representations. The intuition behind such an idea is, since we could reconstruct the entire network with this low dimensional representation, any downstream machine learning task applied to this low dimensional representation is equivalent in applying to the entire network.

4.2 Experimental setup

For implementation[1] we used python 3.7 and chose keras as deep learning library. We used keras custom layer feature to build layers capable of weight tying. We considered *adam* as optimizer and *relu* as activaton function. As loss function we chose *sigmoid-crossentropy* and added 20% dropout to each layer. We trained each autoencoder to atmost 50,000 iterations to reduce the reconstruction error. For each network, a layer configuration that fits in 2 layer architecture has been chosen, which in turn gave flexibility of adopting our model to networks of different ranges. For a particular network, a one-step training of stacked layer

[1] https://github.com/dilberdillu/community-detection-DL

Dataset	Layer Configuration	N	M	K	SP	FUA	FN	DNR	This Model
Polbooks	105-64-32	105	441	3	0.561	0.574	0.531	0.582	**0.600**
Polblogs	1490-256-128	1490	16718	2	0.511	0.375	0.499	0.517	**0.533**
Dolphin	62-32-16	62	159	4	0.753	0.516	0.572	0.818	**0.830**
Football	115-64-32	115	613	12	0.334	0.890	0.698	**0.914**	0.904

Table 1. Normalized Mutual Information of models on real world networks

autoencoder is done instead of layerwise training of multiple autoencoders as done by (Yang et al., 2016). This in fact reduces training time and effort. Figure 2 depicts the gradual decrease of loss with respect to number of epochs for all datasets except *Polblogs*, in which the loss decreases expoentially.

5 RESULTS AND EVALUATION

We compared our model with DNR (Yang et al., 2016), a deep learning approach, and the other existing community detection methods like SP (E.J. Newman, 2006a), FUA (Blondel et al., 2008), FN (E J Newman, 2004). Our model was tested on 4 benchmark datasets (E.J. Newman, 2006b; A. Adamic, 2005; Girvan and E.J. Newman, 2001; Lusseau and Newman, 2004), and found out that it has improved results in 3 of them while a competing result with fourth one. We used NMI as quality measure to understand obtained cluster correlation. In Table 1, second column contains layer configuration of corresponding datasets, N and M refer to the number of nodes, edges respectively and K is the ground truth number of communities that we have used in the experiment. From Table 1, it can be seen that for the datasets of *Polbooks*, *Polblogs* and *Dolphin* networks, the proposed model performs with slightly better improved results and we have a close margin on the *Football* network with DNR(Yang et al., 2016). We believe that reason for this improved result is essentially credited to optimized parameter sharing.

6 CONCLUSION

In this paper we proposed an improved method that solves the problem of community detection. Unlike existing methods that uses autoencoders, we use shared weights across layers and followed a common 2 layer architecture with one-step layer stacked training. Experiments on multiple datasets have shown that the proposed model performs better than the existing state of the art methods in community detection. Our work delivers a convenient framework with a flexibility of adopting the model to networks of different sizes with reduced training time for community detection. For future work, we plan to extend our model and apply it on larger datasets. Our focus will be on scaling as well as improving the model at the same time.

REFERENCES

A. Adamic, L. (2005). The political blogosphere and the 2004 u.s. election: Divided they blog. *Proceedings of the 3rd International Workshop on Link Discovery* doi:10.1145/1134271.1134277

Blondel, V., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics Theory and Experiment* 2008. doi:10.1088/1742-5468/2008/10/P10008

E. Hinton, G. and S. Zemel, R. (1994). Autoencoders, minimum description length and helmholtz free energy. *Advances in Neural Information Processing Systems* 6

E J Newman, M. (2004). Fast algorithm for detecting community structure in networks. *Physical review. E, Statistical, nonlinear, and soft matter physics* 69, 066133. doi:10.1103/PhysRevE.69.066133

E.J. Newman, M. (2006a). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America* 103, 8577–82. doi:10.1073/pnas.0601602103

E.J. Newman, M. (2006b). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America* 103, 8577–82. doi:10.1073/pnas.0601602103

Girvan, M. and E.J. Newman, M. (2001). Community structure in social and biological networks. *proc natl acad sci* 99, 7821–7826

Grover, A. and Leskovec, J. (2016). Node2vec: Scalable feature learning for networks. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA: ACM), KDD '16, 855–864. doi:10.1145/2939672.2939754

Hamilton, W. L., Ying, Z., and Leskovec, J. (2017). Representation learning on graphs: Methods and applications. *IEEE Data Eng. Bull.* 40, 52–74

He, D., Liu, D., Jin, D., and Zhang, W. (2015). A stochastic model for detecting heterogeneous link communities in complex networks. In *AAAI*

Jia, Y., Zhang, Q., Zhang, W., and Wang, X. (2019). Communitygan: Community detection with generative adversarial nets

Jin, D., Chen, Z., He, D., and Zhang, W. (2015). Modeling with node degree preservation can accurately find communities. In *AAAI*

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *CoRR* abs/1609.02907

Lusseau, D. and Newman, M. (2004). Identifying the role that animals play in their social networks. *Proceedings. Biological sciences* 271 Suppl 6, S477–81

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). Deepwalk: Online learning of social representations. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* doi:10.1145/2623330.2623732

Psorakis, I., Roberts, S., Ebden, M., and Sheldon, B. (2011). Overlapping community detection using bayesian non-negative matrix factorization. *Physical review. E, Statistical, nonlinear, and soft matter physics* 83, 066114. doi:10.1103/PhysRevE.83.066114

Vu Tran, P. (2018). Learning to make predictions on graphs with autoencoders. 237–245. doi:10.1109/DSAA.2018.00034

Yang, L., Cao, X., He, D., Wang, C., Wang, X., and Zhang, W. (2016). Modularity based community detection with deep learning