Unsupervised Community Detection with Modularity-Based Attention Model

Ivan Lobov, Sergey Ivanov
Criteo
Paris, France
{i.lobov,s.ivanov}@criteo.com

Abstract

In this paper we take a problem of unsupervised nodes clustering on graphs and show how recent advances in attention models can be applied successfully in a ”hard” regime of the problem. We propose an unsupervised algorithm that encodes Bethe Hessian embeddings by optimizing soft modularity loss and argue that our model is competitive to both classical and Graph Neural Network (GNN) models while it can be trained on a single graph.

1 Introduction

Community detection on graphs is a task of learning similar classes of vertices from the network's topology. It is one of the central problems in data mining and has found numerous applications in sociological studies (Goldenberg et al. (2010)), DNA 3D folding (Cabreros et al. (2016)), product recommendations (Clauset et al. (2004)), natural language processing (Ball et al. (2011)) and more. It is not surprising that many fundamental results have been obtained in recent years that shed the light on our understanding of the solvability of this problem in general. Specifically, precise phase transition and fundamental limits exist for one of the central graph generation models to test community detection algorithms, the so-called Stochastic Block Model SBM(n, p, W) and its symmetric variant *Symmetric Stochastic Block Model*, SSBM(n, k, A, B) (Abbe (2018)). In this paper, we'll be dealing with a simpler case, SSBM.

The optimization of log-likelihood and modularity are equivalent in SSBM model (Newman, 2013). Hence, we propose to use a generalization of modularity that outputs the probability of each cluster for a node and is therefore differentiable for a neural network model. Our model is based on the recent advances in NLP where Transformer model (Vaswani et al. (2017)) with attention mechanism shows superior results. We adopt the encoder part of Transformer to transform initially obtained Bethe Hessian embeddings and then produce the probability of each cluster for each node to optimize with our loss function.

Our contributions include:

- A new model with attention mechanism that optimizes a soft modularity loss function (and release the code to reproduce our results[1]).
- A comparative study of classical algorithms and recent supervised graph neural network and outline several advantages of the unsupervised model.

2 Related Work

SBM models have seen a resurged interest recently due to the conjecture of (Decelle et al. (2011)) that hypotheses phase transition for weak recovery case (the one we are interested in this paper) and the existence of the information-computation gap for 4 communities in the symmetric case. It was proved in (Massoulié (2014); Bordenave et al. (2015); Mossel et al. (2018)) that efficient

[1] https://github.com/Ivanopolo/modnet

algorithms exists when there are 2 communities and signal-to-noise ratio is greater than one (KS threshold), while (Mossel et al. (2015)) shows that it is impossible to detect communities below this threshold for 2-community case. For more than 3 communities it was also proved that there are non-efficient algorithms that can weakly recover communities strictly below KS threshold (Abbe & Sandon (2015)). An extensive overview of the area can be found in Abbe (2018).

Graph neural networks have been recently applied to solve community detection problem with the current state-of-the-art LGNN model(Chen et al. (2019)) designed to extract high-order node interactions via Non-backtracking operator. In parallel line of work, there are graph neural models based on attention mechanisms (Veličković et al. (2018); Kool et al. (2019b)) that have showed prominent results in other domains such as speech recognition and NLP.

3 SSBM Regimes

Signal-to-Noise Ratio (SNR) for SSBM(n, k, A, B) model is defined as:

$$\text{SNR} = \frac{(a-b)^2}{k(a+(k-1)b)} \tag{1}$$

It is known that for $k \geq 2$ and SNR > 1 it is possible to detect communities in polynomial-time (Abbe & Sandon (2016)). Moreover, when SNR ≤ 1 and $k = 2$ the weak recovery of SSBM model is not possible theoretically. However, it is also known that for $k \geq 4$ it is possible to weakly recover in the setup of SNR $> \alpha$, where $\alpha < 1$. This means that there exists a *computational threshold* after which only information-theoretical algorithms can recover communities. This gap between what is possible to compute efficiently and what can be computed in general is known as *information-computation gap*. In this work we compare two different regimes: one regime that lies within a computation threshold and is achievable by classical algorithms (associative case) and one regime for the information-theoretical gap (disassociative case).

4 Approach

4.1 Loss

If we know the generative model from which a graph is produced, it is possible to make statistical inference and evaluate the fitness of the estimated parameters using model's log likelihood. This approach assumes a hard labelling of the nodes and therefore requires some form of relaxation to obtain a differentiable loss function. It has been shown in (Newman, 2013) that optimizing log-likelihood for the SSBM model is equivalent to the modularity optimization, a popular heuristic approach to graph clustering. And there exists (Havens et al., 2013) a generalization of modularity to soft labelling which is well differentiable and can be directly used for learning.
We take a soft modularity M as our loss function:

$$M = -tr(UBU^T)/||A||_1$$

Where $U \in \mathbb{R}^{N \times C}$ is a matrix of probabilities of nodes attribution to clusters (the number of which is a model hyperparameter), $B = A - dd^T/||A||_1$, d is a vector of node degrees, A is the adjacency matrix and $||A||_1 = \sum |A_{ij}|$.
As the modularity might have multiple local optima, we found that it is beneficial if we can include additional information about the graph as a prior to steer the model in the right direction. We use a regularizer defined by:

$$R = \left(\sum_i^C \left(\sum_j^N U_{ij} - \frac{1}{C} \right)^2 \right)$$

Where C is the number of communities in the graph. The regularizer forces the model to find communities of similar sizes as is expected in SSBM.

Our final loss is:

$$\begin{cases} L_a = M + \lambda R, & \text{for associative communities} \\ L_d = -M + \lambda R, & \text{for disassociative communities.} \end{cases} \tag{2}$$

We use negation of modularity loss M as modularity is negative for disassociative case. The benefits of using this loss function are the following: we do not require explicit labels for learning, the loss is invariant under labels permutation and it is a consistent loss under SSBM with soft community partitioning.

4.2 Model

Our network takes as input initialized node embeddings and consists of three main blocks: projection into higher dimensions (done via a fully connected network with skip connections (He et al., 2016) and batch norm (Ioffe & Szegedy, 2015)), graph attention block and output prediction layer with a fully connected layer and a softmax.

Node Embeddings We initialize node embeddings with eigenvectors recovered as part of the Bathe Hessian decomposition, a baseline algorithm described in Section 5.3.

Graph Attention At every layer the attention is done over all neighbors of the node. The Encoder module is described in (Kool et al., 2019a) which is a form of multi-headed attention on neighbors of a node. It is particularly well-suited for our initialization of the node embeddings; the key-value dot-products give the model access to the community structure estimated by the Bethe Hessian eigenvectors.

4.3 Training

Our model is an unsupervised model in a sense that it does not require any labels from the original graph. We train the model separately on each graph repeatedly trying to improve the soft modularity loss. Since the method does not require a larger training set, it significantly reduces the training time.
For all the experiments reported, we trained a model with 2 layers, 3 heads of attention, the size of all hidden layers equal to 48 and regularization parameter $\lambda = 0.5$.

5 Experiments

5.1 Datasets

We generate two datasets SBM(n, k, a/n, b/n) that correspond to associative and disassociative cases. For the disassociative dataset, we set $a = 0$ and $b = 18$, which means that there are no links between vertices of the same cluster. For the associative dataset, we set $a = 21$ and $b = 2$. All datasets have $n = 400$ vertices and $k = 5$ communities. The parameters for the disassociative case correspond to the information-theoretical gap, while for associative case they correspond to the regime when belief propagation can weakly recover the clustering efficiently. We generate 6000 graphs for training dataset and 1000 for validation.

5.2 Evaluation Metrics

Overlap measures the intersection of the original assignment y and the predicted assignment $\hat{y}$ across all possible permutations from the permutation group S (Decelle et al. (2011)):

$$O(y, \hat{y}) = \max_{\pi \in S_{\hat{y}}} \frac{(1/n) \sum_u \delta_{y(u), \pi(\hat{y}(u))}) - 1/c}{1 - 1/c}$$

Where n is the number of nodes in a graph, c is the number of communities and $\delta_{u,v}$ equals one when $u = v$ and zero, otherwise.

Mutual information is a similarity measure between two labels assignments y and $\hat{y}$:

$$I(y,\hat{y}) = \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{y(i)\cap\hat{y}(j)}{n} \log n \frac{y(i)\cap\hat{y}(j)}{|y(i)||\hat{y}(j)|} \tag{3}$$

We use a normalized version (Normalized Mutual Information, hence NMI) of (3) by the arithmetic mean of the entropy for each assignment.

Modularity is a measure of fitness of our inference assuming the SSBM generative model (it differs by a constant factor from the log likelihood). It is positive for associative communities and negative for disassociative ones.

5.3 Baseline Methods

We compare our model with a state-of-the-art GNN model, LGNN, and several classical baselines.

Bethe Hessian, introduced in Saade et al. (2014), is a spectral approximation of Belief Propagation (BP), a classical algorithm for community detection, by employing the Bethe Hessian operator:

$$H = (r^2 - 1)\mathbb{1} - rA + D,$$

where r is the largest eigenvalue of non-backtracking operator B, A is the adjacency matrix and D is a diagonal matrix of degrees. We find the k smallest algebraically eigenvectors of H and find clusters using k-means algorithm. We found that its results are more stable and often better in practice than of BP.

Louvain Modularity (Blondel et al., 2008) is a popular baseline that greedily updates the clusters if the modularity is improved. Since it does not control the number of clusters and we often ended up with more clusters that in the original graph, we did not compute the overlap metric for it (which requires the same number of communities). We do not report results for the disassociative case as the algorithm cannot find communities in this setting by design, it can only merge nodes that are neighbors.

LGNN Chen et al. (2019) is a supervised community recovery algorithm based on GNNs and that can learn operators of inter-graph node and edge connectivity similar to laplacian for nodes or non-backtracking operator for edges.

GNN Kipf & Welling (2017) In addition to the attention model, we also experiment with GNN architecture that produces network embeddings that are trained using our loss equation (2).

For our GNN and Attention models, we also have a choice of initialization of embeddings. We add results for Random and Bethe-Hessian (BH) initialization.

5.4 Empirical Results

Results are presented in Tables 1 and 2. Note that in disassociative case, the smaller value for modularity, the better; for all other metrics, the higher, the better. True labels denote the optimal values for the datasets. Among our 4 proposed algorithms (at the bottom), Attention model with Bethe-Hessian initialization is superior, showing that both the structure of the model and the initialization of embeddings improve the quality on all metrics. Overall, we can see that our modularity-based approach achieves competitive results for the associative and disassociative dataset even compared with supervised LGNN. Our model Attention-BH also achieves the top performance on modularity metric among all algorithms as soft modularity is explicitly present in the loss function. We can see that the loss function that we propose allows us to learn as good or even better fit for the generative model (SSBM), which shows that it can be efficiently used to find communities in a fully unsupervised way, learning only on the current graph.

6 Conclusion

In this work we propose a novel approach on how to recover communities in unsupervised way and conduct experiments comparing to state-of-the-art supervised neural models and unsupervised classical algorithms. Supervised models achieves the state-of-the-art results but with the price of having more parameters and longer training time. We believe it is interesting to combine LGNN architecture with an unsupervised modularity loss as potential future work.

Table 1: Associative Dataset.

Table 2: Disassociative Dataset.

Algorithm	Modularity	Overlap	NMI	Modularity	Overlap	NMI
True Labels	0.52	1.0	1.0	-0.20	1.0	1.0
LGNN (supervised)	0.50	0.80	0.67	-0.16	0.24	0.13
Bethe Hessian	**0.52**	**0.85**	**0.69**	**-0.15**	**0.21**	**0.11**
Louvain Modularity	0.48	N/A	0.48	N/A	N/A	N/A
GNN-Random	0.47	0.53	0.48	0	0.02	0.01
GNN-BH	0.49	0.66	0.61	-0.16	0.2	0.1
Attention-Random	0.42	0.36	0.27	-0.17	0.12	0.04
Attention-BH	**0.51**	**0.78**	**0.67**	**-0.18**	**0.22**	**0.11**

REFERENCES

E. Abbe and C. Sandon. Crossing the ks threshold in the stochastic block model with information theory. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 840–844, 2016.

Emmanuel Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018.

Emmanuel Abbe and Colin Sandon. Detection in the stochastic block model with multiple clusters: proof of the achievability conjectures, acyclic bp, and the information-computation gap. *CoRR*, abs/1512.09080, 2015.

Brian Ball, Brian Karrer, and M. E. J. Newman. An efficient and principled method for detecting communities in networks. *CoRR*, abs/1104.3590, 2011.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008 (10):P10008, 2008.

Charles Bordenave, Marc Lelarge, and Laurent Massoulie. Non-backtracking spectrum of random graphs: Community detection and non-regular ramanujan graphs. In *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, FOCS '15, 2015.

I. Cabreros, E. Abbe, and A. Tsirigos. Detecting community structures in hi-c genomic data. In *2016 Annual Conference on Information Science and Systems (CISS)*, pp. 584–589, 2016.

Zhengdao Chen, Lisha Li, and Joan Bruna. Supervised community detection with line graph neural networks. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=H1g0Z3A9Fm`.

Aaron Clauset, M. E. J. Newman, , and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 2004.

Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborova. Phase transition in the detection of modules in sparse networks. 2011.

Anna Goldenberg, Alice X. Zheng, Stephen E. Fienberg, and Edoardo M. Airoldi. A survey of statistical network models. *Found. Trends Mach. Learn.*, 2(2), 2010.

Timothy C Havens, James C Bezdek, Christopher Leckie, Kotagiri Ramamohanarao, and Marimuthu Palaniswami. A soft modularity function for detecting fuzzy communities in social networks. *IEEE Transactions on Fuzzy Systems*, 21(6):1170–1175, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Wouter Kool, Herke van Hoof, and Max Welling. Attention, learn to solve routing problems! In *International Conference on Learning Representations*, 2019a. URL `https://openreview.net/forum?id=ByxBFsRqYm`.

Wouter Kool, Herke van Hoof, and Max Welling. Attention, learn to solve routing problems! In *International Conference on Learning Representations*, 2019b. URL `https://openreview.net/forum?id=ByxBFsRqYm`.

Laurent Massoulié. Community detection thresholds and the weak ramanujan property. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*, STOC '14, 2014.

Elchanan Mossel, Joe Neeman, and Allan Sly. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 2015.

Elchanan Mossel, Joe Neeman, and Allan Sly. A proof of the block model threshold conjecture. *Combinatorica*, 2018.

Mark EJ Newman. Spectral methods for community detection and graph partitioning. *Physical Review E*, 88(4):042822, 2013.

Alaa Saade, Florent Krzakala, and Lenka Zdeborová. Spectral clustering of graphs with the bethe hessian. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018.