# Generalization Guarantee of Training Graph Convolutional Networks with Graph Topology Sampling

Hongkang Li [1], Meng Wang [1], Sijia Liu[2,3], Pin-Yu Chen[3], Jinjun Xiong[4]

[1]Rensselaer Polytechnic Institute

[2]Michigan State University

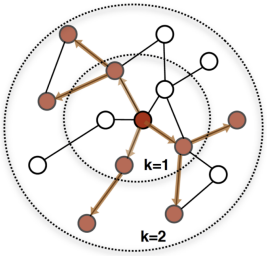
[3]IBM Research

[4]University at Buffalo

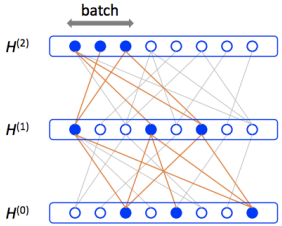
International Conference on Machine Learning (ICML 2022)
July, 2022

# Sampling in Graph Neural Network

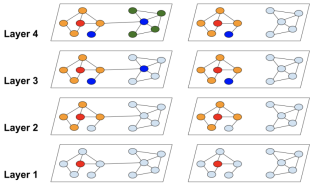
- Graph topology sampling: node-wise, layer-wise, subgraph sampling.
- Why sampling? To reduce computational & memory costs.




*GraphSage (Hamilton et al., 2017)*




*FastGCN (Chen et al., 2018)*




*Cluster-GCN (Chiang et al., 2019)*

*Under what conditions does a graph convolutional network (GCN) learned with graph topology sampling achieve satisfactory generalization?*

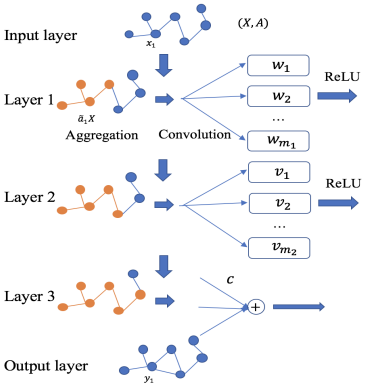
# Problem formulation and GCN Model

Consider a graph with $N$ nodes and the normalized adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{N\times N}$. We study a semi-supervised node classification problem with $|\Omega|$ known labels, where $\Omega \subset [N]$. The goal is to predict unknown labels in $[N]/\Omega$.

- Feature matrix: $\boldsymbol{X} \in \mathbb{R}^{N\times d}$, distribution-free.
- Learner network: three-layer GCN with $m_1$, $m_2$ neurons in the two hidden layers.

$$F_{\boldsymbol{A}}(\boldsymbol{e}_g, \boldsymbol{X}; \boldsymbol{W}, \boldsymbol{V}) = \boldsymbol{e}_g^\top \boldsymbol{A} \cdot \mathrm{ReLU}(\boldsymbol{r} + \boldsymbol{B}_2)\boldsymbol{C}, \quad \boldsymbol{r} = \boldsymbol{A} \cdot \mathrm{ReLU}(\boldsymbol{AXW} + \boldsymbol{B}_1)\boldsymbol{V} \tag{1}$$

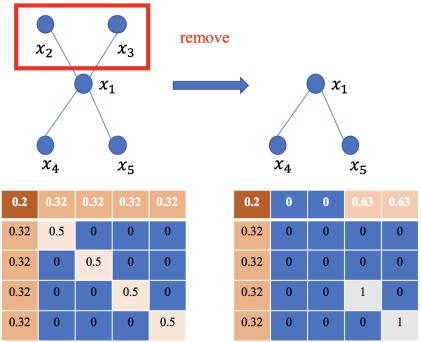
- Loss function: non-negative & convex, including $\ell$-2 regression and cross entropy
- SGD with graph topology sampling.



# Graph topology Sampling

Categorize the nodes of the graph into different groups based on orders of degree. Implement group-wise uniform sampling.

- Randomly removing nodes and the incident edges.
- Higher sampling rate on higher-degree nodes.
- For sampled nodes, scale the corresponding columns of $\boldsymbol{A}$. For unsampled nodes, set the corresponding columns of $\boldsymbol{A}$ to 0.




*Effective adjacency matrix* $\boldsymbol{A}^*$: $\boldsymbol{A}^* = \boldsymbol{A} \cdot \mathrm{diag}(p_1^*, p_2^*, \cdots, p_N^*)$, where $p_i^*$, $i \in [N]$ is the group-wise sampling probability of node $i$. $\boldsymbol{A}^*$ is more balanced than $\boldsymbol{A}$ for unbalanced graphs.

# Theorem 1 (informal)

For any small $\gamma,\ \epsilon > 0$, as long as the overparameterization satisfies

$$m_1 = m_2 = m \geq \text{poly}\Big(\|\boldsymbol{A}^*\|_\infty, \epsilon^{-1}\Big), \tag{2}$$

and the sample complexity satisfies

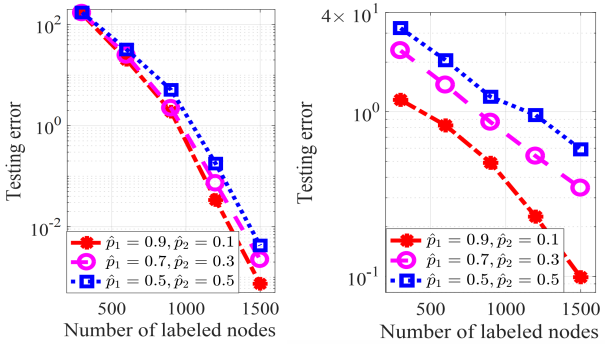
$$|\Omega| \geq \Theta(\text{poly}(\epsilon^{-1}, \|\boldsymbol{A}^*\|_\infty) \cdot \log N \log m), \tag{3}$$

- Training with graph topology sampling returns a model that has the same performance as the model trained by a GCN with $\boldsymbol{A}^*$ as the *effective* adjacency matrix
- A generalization error at most $(1+\gamma)\text{OPT}_{\boldsymbol{A}^*} + \epsilon$.
- $\text{OPT}_{\boldsymbol{A}^*}$: the smallest population risk over the choices of $\boldsymbol{C}^*, \boldsymbol{W}_1^*, \boldsymbol{W}_2^*, \boldsymbol{V}_1^*, \boldsymbol{V}_2^*$ in the target function, which is defined as

$$F_{\boldsymbol{A}^*}^*(\boldsymbol{e}_g, \boldsymbol{X}) = \boldsymbol{e}_g^\top \boldsymbol{A}^* \big(\Phi(\boldsymbol{r}_1) \odot \boldsymbol{r}_2\big) \boldsymbol{C}^*,\ \boldsymbol{r}_1 = \boldsymbol{A}^* \phi_1(\boldsymbol{A}^* \boldsymbol{X} \boldsymbol{W}_1^*) \boldsymbol{V}_1^*,\ \boldsymbol{r}_2 = \boldsymbol{A}^* \phi_2(\boldsymbol{A}^* \boldsymbol{X} \boldsymbol{W}_2^*) \boldsymbol{V}_2^* \tag{4}$$

# Numerical Results

- Graph sampling reduces the impact of dominating nodes, resulting in a more balanced $\boldsymbol{A}^*$.
- Similar results of FastGCN (Chen et al., 2018).



# References


References

- Hamilton, W., Ying, Z., and Leskovec, J. Inductive representation learning on large graphs. In Advances in neural information processing systems, pp. 1024-1034, 2017.
- Chen, J., Ma, T., and Xiao, C. Fastgcn: Fast learning with graph convolutional networks via importance sampling. In International Conference on Learning Representations, 2018
- Chiang, W.-L., Liu, X., Si, S., Li, Y., Bengio, S., and Hsieh, C.-J. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery  Data Mining, pp. 257-266, 2019.