ALTERNATELY OPTIMIZED GRAPH NEURAL NETWORKS

Zepeng Zhang

June 16, 2022

In practice, the end-to-end training for decoupled GNNs can be inefficient in computation and memory consumption, in that gradients need to backpropagate through all feature aggregation layers and all the hidden states in aggregation layers need to be stored. This paper [1] proposes an efficient alternating optimization framework for decoupled GNNs.

In the end-to-end training of decoupled GNNs, there are three steps, i.e., feature transformation, feature aggregation, and loss function. In this paper, instead of using an end-to-end way, they propose a general formulation using a latent variable $\mathbf{F}$ to capture these three types of information:

$$\underset{\mathbf{F},\theta}{\arg\min}\mathcal{L} = \lambda_1\mathcal{D}_1\left(f_\theta(\mathbf{X}), \mathbf{F}\right) + \mathcal{R}(\mathbf{F}, \mathbf{A}) + \lambda_2\mathcal{D}_2(\mathbf{F}, \mathbf{Y}),$$

where f_θ is the feature transformation function, $\mathcal{D}_1$ and $\mathcal{D}_2$ are the distance metrics that measure the similarity of two components, and $\mathcal{R}$ is a regularization to constrain $\mathbf{F}$ by the graph structure information. We can treat $\mathbf{F}$ as a kind of soft pseudo label matrix and then (1) the first term is to map the node features to their corresponding pseudo labels; (2) the second term is to indicate that $\mathbf{F}$ should follow some properties guided by the graph structure information; (3) the third term constrains that $\mathbf{F}$ should be close to the ground-truth labels. Note that we can set $\lambda_2 = 0$ for those unlabeled nodes.

An instance of the general alternately optimized GNN (ALT-OPT) framework is proposed:

$$\mathcal{L} = \lambda_1\|\mathrm{MLP}(\mathbf{X}) - \mathbf{F}\|_F^2 + \mathrm{tr}\left(\mathbf{F}^\top\tilde{\mathbf{L}}\mathbf{F}\right) + \lambda_2\|\mathbf{F} - \mathbf{Y}\|_F^2.$$

Due to the coupling between $\mathbf{F}$ and MLP, it is difficult to find optimal solutions for both $\mathbf{F}$ and MLP. In this work, an alternating optimization scheme is adopted.

Update F. Fixing MLP, we can minimize $\mathcal{L}$ w.r.t. $\mathbf{F}$ using gradient descent, which gives

$$\begin{aligned}\mathbf{F}^{k+1} &= \mathbf{F}^k - \eta\frac{\partial\mathcal{L}}{\partial\mathbf{F}}\\ &= \mathbf{F}^k - 2\eta\left(\lambda_1(\mathbf{F} - \mathrm{MLP}(\mathbf{X})) + \tilde{\mathbf{L}}\mathbf{F} + \lambda_2(\mathbf{F} - \mathbf{Y})\right)\\ &= 2\eta\left(\tilde{\mathbf{A}}\mathbf{F}^k + \lambda_1\mathrm{MLP}(\mathbf{X}) + \lambda_2\mathbf{Y}\right) + \left(1 - 2\eta\left(\lambda_1 + \lambda_2 + 1\right)\right)\mathbf{F}^k.\end{aligned}$$

According to the smoothness and strong convexity of the problem, we can set $\eta = \frac{1}{2(\lambda_1+\lambda_2+1)}$ and the update formula becomes

$$\mathbf{F}^{k+1} = \frac{1}{\lambda_1 + \lambda_2 + 1}\tilde{\mathbf{A}}\mathbf{F}^k + \frac{\lambda_1}{\lambda_1 + \lambda_2 + 1}\mathrm{MLP}(\mathbf{X}) + \frac{\lambda_2}{\lambda_1 + \lambda_2 + 1}\mathbf{Y}.$$

Since $\mathbf{F}$ acts as pseudo labels when training the MLP, it is further normalized using softmax function. After training, $\mathbf{F}$ is used for prediction.

Update MLP. Fixing $\mathbf{F}^{k+1}$, we train MLP by minimizing $\|\mathrm{MLP}(\mathbf{X}) - \mathbf{F}\|_F^2$ using all the labeled nodes and a same number of unlabeled nodes for each class with the highest weight. The weight for each unlabeled node is computed as $w_i = 1 - \frac{H(\mathbf{F}_i)}{\log(C)} \in [0, 1]$ where $H\left(\mathbf{F}_i\right) = -\sum_{j=1}^{C}\mathbf{F}_{ij}\log\mathbf{F}_{ij}$ is the entropy of the pseudo label $\mathbf{F}_i$.

To get a good initialization of MLP, the feature matrix is first preprocessed by solving a graph signal denoising problem. Then the MLP is trained using the labeled data for a few epochs to get an initialization.

The experiment results verify that ALT-OPT is more efficient and performs especially well when the label rate is low. Besides, ALT-OPT converges faster than APPNP and can achieve the same results as APPNP with fewer layers.

Moreover, using the trained MLP in ALT-OPT as the initialization for the MLP in APPNP can enhance the performance of APPNP.

References

[1] Haoyu Han, Xiaorui Liu, Torkamani Ali, Feng Shi, Victor Lee, and Jiliang Tang. Alternately optimized graph neural networks. *arXiv preprint arXiv:2206.03638*, 2022.