Supplementary material: On Dropping Clusters to Regularize Graph Convolutional Neural Networks

Xikun Zhang, Chang Xu, and Dacheng Tao

UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering, The University of Sydney, Darlington, NSW 2008, Australia

1 Parallelized computation of correlation matrix M_{corr}

Algorithm 1: Correlation matrix computation

Input: $X \in \mathbb{R}^{N\times d}$
Output: $M_{corr} \in \mathbb{R}^{d\times d}$
Compute the transpose of X: $X^T = Transpose(X) \in \mathbb{R}^{d\times N}$
Covariance computation:
  $E' = \frac{X^T \cdot X}{N} \in \mathbb{R}^{d\times d}$
  $E = Mean(X^T, dimension = 1) \in \mathbb{R}^{d\times 1}$
  $M = E \cdot E^T \in \mathbb{R}^{d\times d}$
  Covariance matrix: $Cov = E' - M$
Mean square of each channel: $E_{X^2} = Mean({X^T}^2, dimension = 1) \in \mathbb{R}^{d\times 1}$, ${X^T}^2$ is element-wise square of X^T
$Var = E_{X^2} - E^2 \in \mathbb{R}^{d\times 1}$
$\sigma = \sqrt{Var} \in \mathbb{R}^{d\times 1}$
$\sigma' = \sigma \cdot \sigma^T \in \mathbb{R}^{d\times d}$
$M_{corr} = \frac{Cov}{\sigma'} \in \mathbb{R}^{d\times d}$
return M_{corr}

2 DropCluster

2.1 Number of seed entries

In this part, we give the detailed computation of Eqs. 9 and 10 in the paper. Given the input $X \in \mathbb{R}^{N\times d}$ and dropping rate r_d, the number of entries to drop is $N \times d \times r_d$. We suppose the number of seed entries is n_{seed}. For the first layer, we select the 1-hop neighbors. With the average number of edges being n_e, each seed entry corresponds to n_e 1-hop neighboring entries on average. Considering the channel correlation, we could formulate:

$$n_{seed} \times (1 + n_e) \times n_c = N \times d \times r_d, \tag{1}$$

i.e.

$$n_{seed} = \frac{N \times d \times r_d}{(1 + n_e) \times n_c} \tag{2}$$

When $l > 1$, besides the 1-hop neighbors, each seed entry also corresponds to 2-hop to l-hop neighbors. Assuming that the 1-hop neighbors of the seed entries do not have common 1-hop neighbors. The number of 2-hop neighbors of a seed entry could be approximated as $n_e \times (n_e - 1)$, where the term $(n_e - 1)$ denotes excluding the seed entries. Similarly, we can approximate the i-hop neighbors of a seed entry as $n_e \times (n_e - 1)^{i-1}$. Thus, for the l-th layer, we could derive:

$$n_{seed} \times (1 + n_e + \sum_{i=2}^{l} n_e \times (n_e - 1)^{i-1}) \times n_c = N \times d \times r_d, \tag{3}$$

i.e.

$$n_{seed} = \frac{N \times d \times r_d}{(1 + n_e + \sum_{i=2}^{l} n_e \times (n_e - 1)^{i-1}) \times n_c}. \tag{4}$$