Supplementary Materials for

A Simple yet Effective Method for Graph Classification

A Three kinds of configurations for experimental results.

To the best of our knowledge, there have been three kinds of configurations for experimental results.

- 10-fold cross-validation with one set of optimal hyperparameters, in which the experimental result is the best cross-validation accuracy averaged over the 10 folds, i.e., $acc = np.array(ACC).mean(axis = 1).max()$, where $ACC \in \mathbb{R}^{Epoch \times Fold}$ is the model accuracy matrix under the same configuration. This configuration is adopted by GIN [Xu *et al.*, 2019] and LP-GNN [Tiezzi *et al.*, 2021].
- 10-fold cross-validation with one set of optimal hyperparameters, in which the experimental result is the average accuracy over the best validation accuracy for each fold, i.e., $acc = np.array(ACC).max(axis = 0).mean()$, where $ACC \in \mathbb{R}^{Epoch \times Fold}$ is the model accuracy matrix under the same configuration. This configuration is adopted by second-order pooling (SOPOOL) [Wang and Ji, 2020].
- 10-fold cross-validation with each fold a set of optimal hyperparameters, in which the experimental result is the average accuracy over the average accuracy for each fold, i.e., $acc = np.array(ACC).mean()$, where $ACC \in \mathbb{R}^{Epoch \times Fold}$ is the model accuracy matrix under the optimal configuration for each fold. This configuration is proposed by [Errica *et al.*, 2020] for graph classification.

In general, the third configuration is the most fair way to compare among methods; however, this configuration requires much more computing power than the other two methods because of its hyperparameter selection for each fold. In this work, based on our choice of baselines and our available computing power, we adopt the first configuration to obtain our experimental results for a fair comparison.

References

[Errica *et al.*, 2020] Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. In *ICLR*, 2020.

[Tiezzi *et al.*, 2021] Matteo Tiezzi, Giuseppe Marra, Stefano Melacci, and Marco Maggini. Deep constraint-based propagation in graph neural networks. *IEEE Transactions on PAMI*, 2021.

[Wang and Ji, 2020] Zhengyang Wang and Shuiwang Ji. Second-order pooling for graph neural networks. *IEEE Transactions on PAMI*, 2020.

[Xu *et al.*, 2019] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *ICLR*, 2019.