We thank all the reviewers for the constructive comments. We've done additional experiments to address the concerns about the comparison with very recent efforts published this year, and clarified the confusions one-by-one. We reiterate our contributions: (1) Global & local metrics; (2) general DGN to improve GNNs even comparing with recent efforts.

To Reviewer 1. Q: Comparison with GCNII*[1]. We explore to relieve the over-smoothing issue from the perspective of group normalization, and propose a general module – DGN. It can improve different GNN backbones including GCNII*. Our experiments focus on the comparison among existing normalization approaches, rather than fine tuning hyperparameters for state-of-the-art performance. The mentioned paper[1] appeared (July 2020) after NeurIPS submission. We will include the comparison results on backbone network GCNII* in the revised version. Table 1 presents part of the results, where DGN consistently outperforms the other normalization approaches for different network depths. NN (none normalization) denotes the original GCNII* model. It could be further improved by DGN.

Table 1: The test accuracy in percentage over backbone network GCNII* on the dataset Cora.

# Layers	NN	BN	PN	DGN	# Layers	NN	BN	PN	DGN	# Layers	NN	BN	PN	DGN
2	80.2	76.5	74.2	**83.0**	16	83.5	74.4	73.3	**85.1**	64	85.3	73.7	68.1	**86.4**

Q: Metrics. (1) Group distance ratio measures the over-smoothing based on the community impact on the preferred node distance, which has not been studied before. This intuitive metric provides us an empirical guidance to design DGN module to improve model performance. (2) As mentioned in Section 2.3, both over-fitting and over-smoothing influence metric G_{Ins} of instance information gain. For SGC model, we project node features into label space in the first layer and remove all the following trainable matrices. The output representation is generated from the final convolutional layer to eliminate the impact of prediction layer. We observe that G_{Ins} still decreases with more layers, which validates the rationality of our metric tailored to measure the mutual information between graph signals. Indeed, GCN/GAT usually transforms features into label space in the final convolutional layer, and removes the redundant prediction layer.

Q: Training process. Thanks for pointing it out. We find that DGN helps GNN models converge to a better training/validation loss quickly within hundreds of epochs. Due to space limit, we will include it in the revised version.

Q: Missing features. The current benchmarks associated informative node attributes usually require shallow GNNs. But the over-smoothing issue only appears in the scenarios necessary to stack deeper GNNs to access useful neighbors many hops away. As discussed in Section 3.3, one of such scenarios, missing features, exists in many real-world applications and has been studied before. Our DGN is shown to effectively tackle this issue and enable deeper neural architectures.

To Reviewer 2. Q: Increased params. Thanks for the comment. In GNNs, the increased parameters may instead lead to over-fitting. As illustrated in Figure 3 and indicated by group distance ratio, the performance improvement of DGN is indeed brought by soft clustering to disentangle unrelated nodes, comparing with the single group in PN & BN.

Q: Deeper vs shallow models. The deeper model unleashes power in the scenario where a node really needs information many hops away to better represent its label. We study such a complex scenario with missing node features in Section 3.3. As also shown in Table 2 and Appendix, deeper GNNs+DGN achieve better performance than the shallow ones.

Q: Softmax. We've tried Softmax temperatures and regularizations to enforce that each node is assigned to one group. We empirically found that soft clustering itself can achieve the similar objective and reduce computation complexity.

To Reviewer 3. Q: Comparing attentive normalization. Thanks for pointing it out, and we will include the comparison in the revised version. Here we present part of results of attentive normalization[2] (AN), attentive context normalization[3] (ACN) and our DGN in Table 2, where DGN consistently outperforms them in all cases. This is because ACN has only one group and AN additionally samples random groups during model inference. They are not in line with the transductive node classification task where the underlying graph has a series of fixed community structures. Furthermore, the motivations for our DGN and AN/ACN are different. While AN/ACN target at capturing the long-range relations between pixels, DGN intends to improve the distance between different groups to mitigate the over-smoothing issue.

Table 2: Test accuracy in percentage over GCN model for dataset Cora.

# Layers	AN	ACN	DGN	# Layers	AN	ACN	DGN	# Layers	AN	ACN	DGN
2	77.1	74.4	**82.0**	15	16.3	42.8	**75.2**	30	17.2	20.9	**73.2**

Q: Zero assignment. A similar case happens in the right part of Figure 3 in our paper, where only 6 out of 10 groups are assigned with nodes. In the hyperparameter study, we further find that this issue could be solved by simply choosing a proper group number close to the class categories. We will test the regularization term in paper[2] in the revised version.

To Reviewer 5. Q: Case studies. We appreciate that you found the work valuable and pointed out this question. As mentioned in Section 2.3, SGC achieves better performance with a larger value of K, since the simplifying mechanism avoids over-fitting to enable layer stacking. Compared with Table 1, Table 2 presents results on the complex scenarios with missing features. As explained in Section 3.3, the deeper models would be preferred since we need to exploit the distant neighbors to collect sparse information. We will explain them in detail in the revised experiment section.

[1] Chen et al. "Simple and Deep Graph Convolutional Networks," ICML 2020. [Available in arXiv on 4 Jul 2020]

[2] Wang et al. "Attentive Normalization for Conditional Image Generation," CVPR 2020. [Available in arXiv on 8 Apr 2020]

[3] Sun et al. "ACNe: Attentive Context Normalization for Robust Permutation-Equivariant Learning," CVPR 2020. [23 Apr 2020]