R5:Official Review of Submission1720 by Reviewer 7Fmz

Q1. In the abstract, the authors argue that current solutions seldom consider both neighbor quality and neighbor quantity. However, it seems that some previous anti-over-smoothing methods that randomly drop nodes or edges will consider both two issues.
Reply:
Drop methods include DropNode, DropEdge, DropMessage, which improve the quality of aggregated neighbour information by randomly dropping operations at each aggregation thus making the variability of aggregated neighbours' information increase. However, on the T-SNE plots of the real dataset we find that although they somewhat prevent excessive smoothing, the performance of the model still decrease rapidly, since they are not concerned about the quantity issue, i.e., tons of neighbour information submerge node individual information.
On the contrary, our TSC approach retains the shallow information of nodes for preventing the loss of node individuality; it also enhances the variability of nodes' information and ensures the quality of information by imposing contrast constraints.

Q2. The concept of neighbor quality and its relationship to over-smoothing is not clear. It would be helpful if the authors could provide more detailed explanations to enhance understanding of neighbor quality.**Reply:**
(1) we can define quality and quantity as follows:
Neighbor quality: If two nodes receive the same or similar information from neighbors, we argue that the neighbor quality of the two nodes is low from the perspective of oversmoothing. This can happen to two neighboring nodes in shallow layers as shown in Figure a1 and a2 (nodes A and B). If two nodes have very low-quality neighbors, they will become similar very quickly (oversmoothing happens). According to the small-world phenomenon [17], any two nodes will become neighbors directly or indirectly by 6 jumps. Therefore, with layers increasing, the neighbor quality will become low between any two nodes as shown in Figures a1 and a2 (nodes A and C)

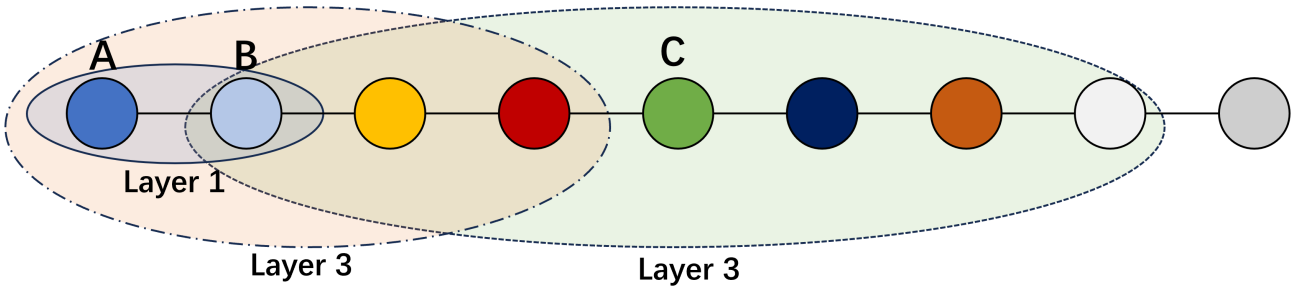


Figure a1. A simplest graph to demonstrate node similarity.

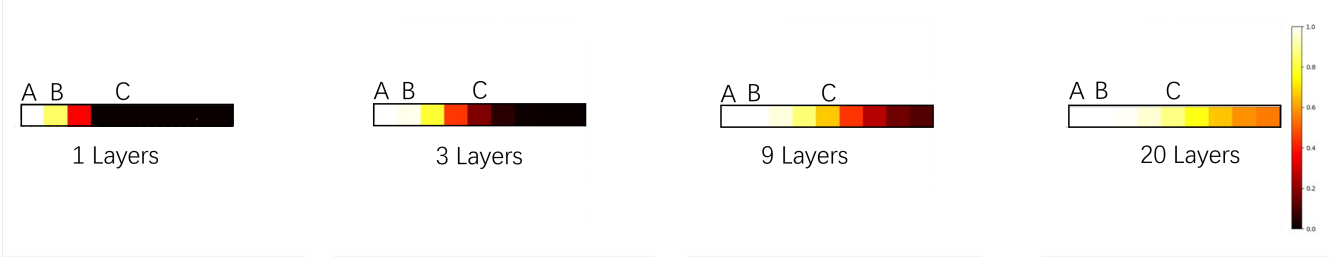
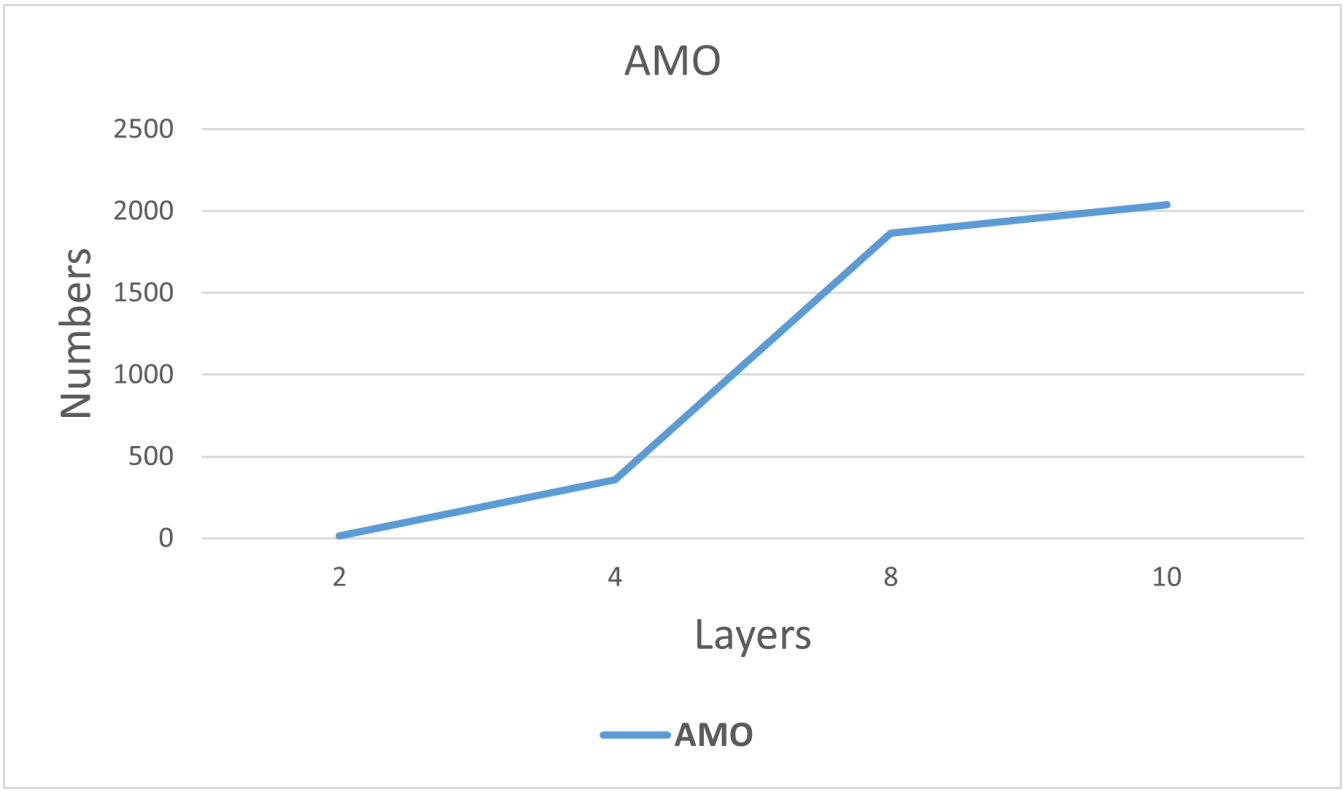


Figure a2. Node similarity changing between A and others. A and B become similar in the 1st layer, because they are neighbor to each other. With layers increasing, A and C becomes similar in layer >3 where they share similar neighbors.

We can use an indicator **Average Mutual Overlapping** (AMO) to describe the neighborhood quality. AMO is defined as follows:

$$S_{i,j} = \begin{cases} 1, A^l_{i,j} > 0 \wedge i \neq j \\ 0, A^l_{i,j} = 0 \end{cases}, Num = \text{Mean}(\text{SS}^T))$$

l is the number of layers and A is the adjacency matrix.
As its name suggests, this indicator averages the number of common neighbors of two nodes over all possible pairs of nodes. It is a sufficient indicator since two nodes have similar aggregation information when they have a large number of common neighbors. On cora, we stack SGC by 2, 4, 8, and 10 layers and measure the AMO. The measurement of the AMO is presented in the Fig. a3.



.

Figure a3. The neighbor quality changeing with respoect to layer numbers

Neighbor quantity: The information received from neighbors grows exponentially, causing nodes to lose their own individuality and accuracy decreases. As shown in Fig.

a1, node C has 6 neighbors in layer 3. If the graph become non-linear, the number of neighbors will be exponential to layer number. The Neighbor quantity can be measured by **Average Number of Different Classes of Nodes in the neighbourhood** :

$$ANDCN = \frac{1}{N}\sum_{i=0j=0}^{N} 1_{[L_i!=L_j \wedge S_{i,j}=1]},$$

where N is the number of nodes, L_i the node i's label,$S_{i,j}$ is defined in AMO.

We plot the variation of $ANDCN$ in layers 2, 4, 8 and 10 on the cora dataset in Figure a4.:

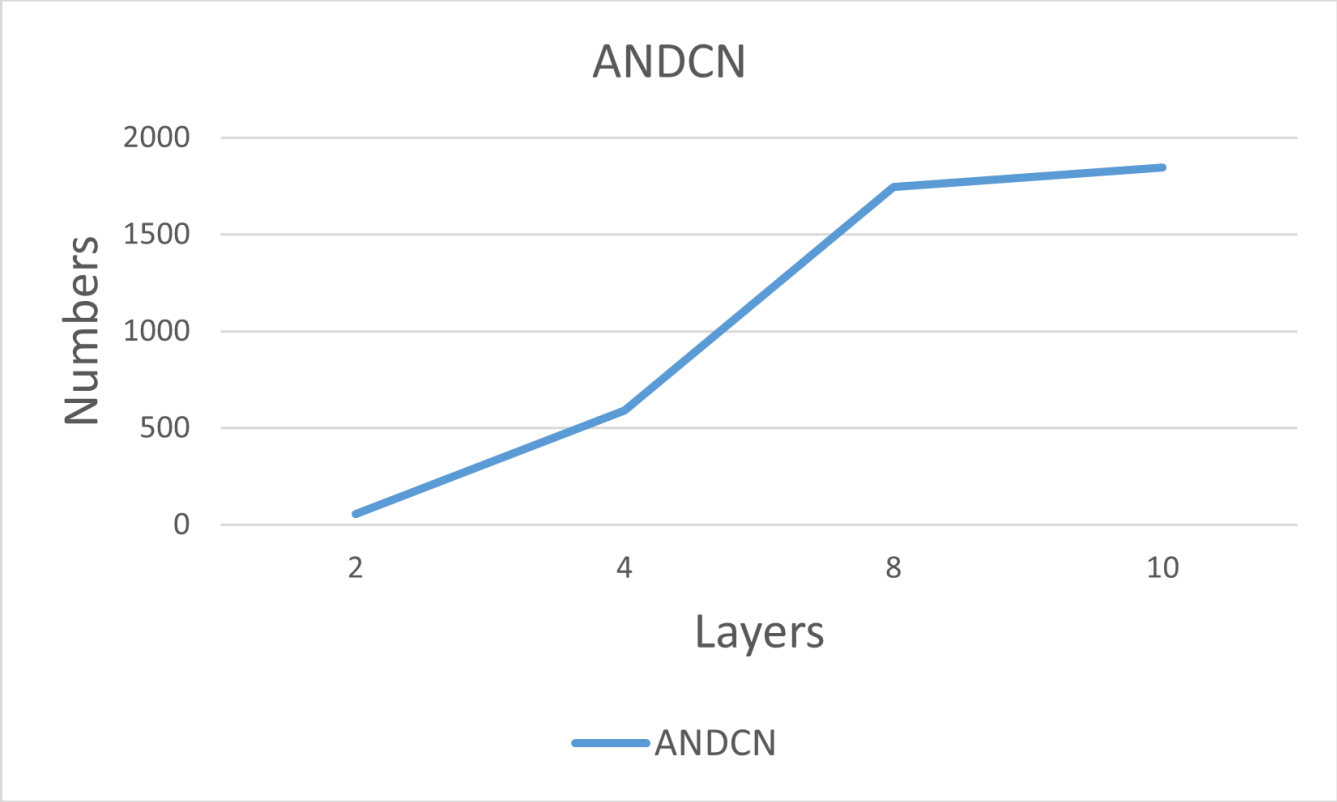


Figure a4. The neighbor quality changeing with respoect to layer numbers

(2) Our motivation can be generally summarized: (a) using random masking to drop a large number of aggregated information and maintain the node's individual information generate from shallow layers. (b) adopting contrastive constraint to enhance node's difference to each other in deep layers.

Q3. In Lines 153-156, why do the authors attribute the failure of DropMessage in the 32-th layer to information from neighbors? In Lines 160-162, why dose the ineffectivity of ContraNorm caused by the overwhelming of neighbors? It requires more explanation to understand the underlying reasons.
Reply: Dropmessage drops a portion of the message at each message passing, but with the number of layers, more and more messages are received, making the node lose its personality.After that the performance of the model gradually decreases.
ContraNorm Although it prevents dimension collapse and thus prevents nodes from converging, as the number of layers increases the amount of messages grows and

nodes lose their individuality.After that the performance of the model gradually decreases.

Q4. In Lines 171-175, the authors claim that some previous deep GCNs might overemphasize shallow layers while ignoring deep layers. However, it is worth noting that approaches such as GCNII [1] and GPRGNN [2] provide mechanisms to control and adaptively tune the importance of node representations from different layers. The authors should consider acknowledging these methods and providing a more nuanced comparison to strengthen their argument.
Reply: GCNII is just a simple summation of the messages at each layer, with a high percentage of shallow messages preserved, and a small percentage preserved for high-level messages, and it inherently does not prevent over-smoothing of high-level messages.
Whereas our approach then preserves the high level messages while using contrast learning to keep the feature distribution spread out and increase the discriminative nature of the nodes.

Q5The source codes and pre-processed datasets are not provided, limiting the reproducibility of this work. Please refer to https://anonymous.4open.science/.

ref
[1] Chen, Ming, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. "Simple and deep graph convolutional networks." In International conference on machine learning, pp. 1725-1735. PMLR, 2020.

[2] Chien, Eli, Jianhao Peng, Pan Li, and Olgica Milenkovic. "Adaptive universal generalized pagerank graph neural network." arXiv preprint arXiv:2006.07988 (2020).