Towards Deeper Graph Neural Networks

作者：Meng Liu，Hongyang Gao，Shuiwang Ji

Texas A&M University

KDD 2020

Abstract

本文对于GNN在使用更深的层时出现的准确率大大下降问题关键因素进行了探究，并且提出了对应的方法，使用了一种自适应深度的GNN。

创新点/贡献/优势：

- 提出了两种独立的操作进行transformation、propagation。
- 自适应的深度学习机制

Introduction

Figure 1: t-SNE visualization of node representations derived by different numbers of GCN layers on Cora. Colors represent node classes.

- 在GNN中使用多层的网络会出现过度平滑的问题（over-smoothing），过度平滑即是不同类的点特征趋近于相同，导致无法分类。
- 出现过度平滑的问题，主要是由于representation transformation 和 propagation的entanglement（纠缠）

$$\begin{aligned} \boldsymbol{a}_i^{(\ell)} &= \text{PROPAGATION}^{(\ell)}\left(\left\{\boldsymbol{x}_i^{(\ell-1)}, \{\boldsymbol{x}_j^{(\ell-1)} | j \in \mathcal{N}_i\}\right\}\right) \\ \boldsymbol{x}_i^{(\ell)} &= \text{TRANSFORMATION}^{(\ell)}\left(\boldsymbol{a}_i^{(\ell)}\right). \end{aligned} \tag{1}$$

Analysis of Deep GNNS

- 定量的分析节点特征的平滑值

$$D(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{2}\left\| \frac{\boldsymbol{x}_i}{\|\boldsymbol{x}_i\|} - \frac{\boldsymbol{x}_j}{\|\boldsymbol{x}_j\|} \right\|, \tag{3}$$

$$SMV_i = \frac{1}{n-1} \sum_{j \in V, j \neq i} D(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{4}$$

$$SMV_G = \frac{1}{n} \sum_{i \in V} SMV_i. \tag{5}$$

- SMV_g 就是整张图的平滑度值，SMV_g 越大，平滑度就越小。

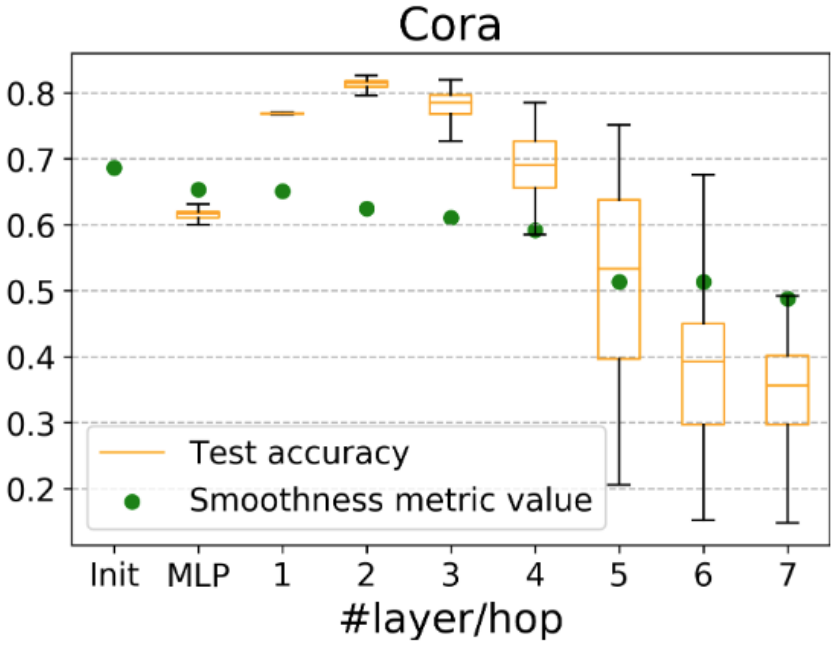


Figure 2: Test accuracy and smoothness metric value of node representations with different numbers of GCN layers on Cora. "Init" means the smoothness metric value of the original data.

可以看到一开始精确值不断提高，但是后面精确值就波动很大，平均值也一直在下降，对于整个图的平滑定量值却略有下降，也就是图平滑度上升

- 一些观点认为过度平滑是由于多次迭代而导致的，但是本文作者提出质疑，他认为对于上图的结果，cora 数据集是很稀疏的，不会因为几次迭代就出现这样精度下降的情况，而且图的平滑度上升并不多。
- 提出关键原因**转化和传播的纠缠极大地损害了深图神经网络的性能。**
  - 特征表式和传播的纠缠使转换中的参数数量与传播中的接收场交织在一起。 传播需要变换函数，因此在考虑大的接收场时会导致大量的参数。 因此，可能很难训练具有大量参数的深层GNN。
  - 特征表式和传播应该式独立的，节点的类别可以通过其初始特征完全可预测，在连接的节点通常属于同一类的假设下，基于图结构的传播可以通过使同一类中的节点表示相似来帮助简化分类任务。 例如，直观上，文档的类别完全由其内容（即，通过单词嵌入来实现）确定，而不是与其他文档的引用关系。 利用其邻居的功能可以简化文档的分类。 因此，表示形式的转换和传播分别从特征和结构方面发挥各自的作用。
- 提出了两个decouple的特征表式和传播操作

$$
\begin{aligned}
Z &= \mathrm{MLP}(X) \\
X_{out} &= \mathrm{softmax}\left(\widehat{A}^{k} Z\right).
\end{aligned} \tag{6}
$$

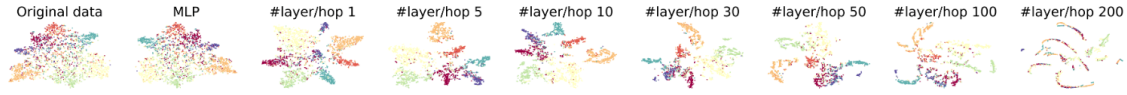


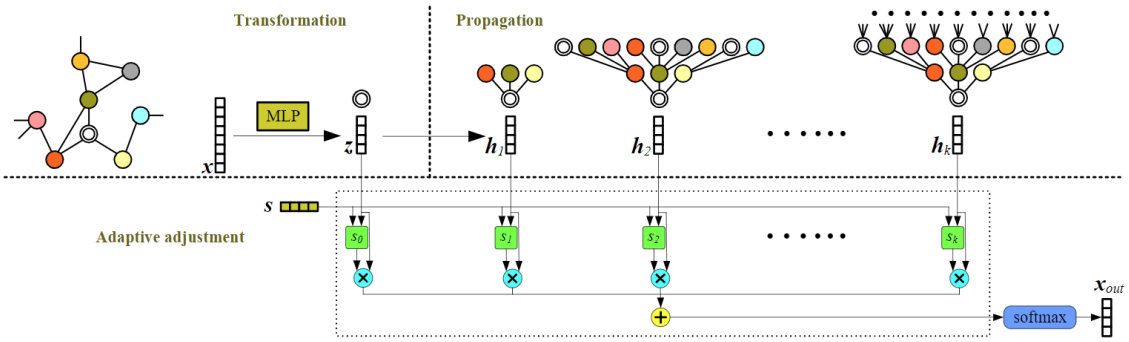
Figure 3: t-SNE visualization of node representations derived by models as Eq.(6) with different numbers of layers on Cora. Colors represent node classes.

Model

- 该模型使特征表示与传播decouples（解耦），从而可以应用较大的接收场而不会导致性能下降。
- 它利用一种自适应调整机制，该机制可以针对每个节点自适应地平衡来自本地和全局邻域的信息，从而导致更具区分性的节点表示形式。

$$
\begin{aligned}
Z &= \mathrm{MLP}(X) && \in \mathbb{R}^{n \times c} \\
H_{\ell} &= \widehat{A}^{\ell} Z, \ell = 1, 2, \cdots, k && \in \mathbb{R}^{n \times c} \\
H &= \mathrm{stack}(Z, H_1, \cdots, H_k) && \in \mathbb{R}^{n \times (k+1) \times c} \\
S &= \sigma(Hs) && \in \mathbb{R}^{n \times (k+1) \times 1} \\
\widetilde{S} &= \mathrm{reshape}(S) && \in \mathbb{R}^{n \times 1 \times (k+1)} \\
X_{out} &= \mathrm{softmax}\left(\mathrm{squeeze}\left(\widetilde{S} H\right)\right) && \in \mathbb{R}^{n \times c},
\end{aligned} \tag{8}
$$

其中s是一个训练的向量，用来调整对于每一层的特征的权重



Experiments

数据集

Dataset	#Classes	#Nodes	#Edges	Edge Density	#Features	#Training Nodes	#Validation Nodes	#Test Nodes
Cora	7	2708	5278	0.0014	1433	20 per class	500	1000
CiteSeer	6	3327	4552	0.0008	3703	20 per class	500	1000
PubMed	3	19717	44324	0.0002	500	20 per class	500	1000
Coauthor CS	15	18333	81894	0.0005	6805	20 per class	30 per class	Rest nodes
Coauthor Physics	5	34493	247962	0.0004	8415	20 per class	30 per class	Rest nodes
Amazon Computers	10	13381	245778	0.0027	767	20 per class	30 per class	Rest nodes
Amazon Photo	8	7487	119043	0.0042	745	20 per class	30 per class	Rest nodes

Models	Cora		CiteSeer		PubMed	
	Fixed	Random	Fixed	Random	Fixed	Random
MLP	61.6 ± 0.6	59.8 ± 2.4	61.0 ± 1.0	58.8 ± 2.2	74.2 ± 0.7	70.1 ± 2.4
ChebNet	80.5 ± 1.1	76.8 ± 2.5	69.6 ± 1.4	67.5 ± 2.0	78.1 ± 0.6	75.3 ± 2.5
GCN	81.3 ± 0.8	79.1 ± 1.8	71.1 ± 0.7	68.2 ± 1.6	78.8 ± 0.6	77.1 ± 2.7
GAT	83.1 ± 0.4	80.8 ± 1.6	70.8 ± 0.5	68.9 ± 1.7	79.1 ± 0.4	77.8 ± 2.1
APPNP	83.3 ± 0.5	81.9 ± 1.4	71.8 ± 0.4	69.8 ± 1.7	80.1 ± 0.2	79.5 ± 2.2
SGC	81.7 ± 0.1	80.4 ± 1.8	71.3 ± 0.2	68.7 ± 2.1	78.9 ± 0.1	76.8 ± 2.6
DAGNN (Ours)	**84.4 ± 0.5**	**83.7 ± 1.4**	**73.3 ± 0.6**	**71.2 ± 1.4**	**80.5 ± 0.5**	**80.1 ± 1.7**

Table 4: Results with different training set sizes on Cora in terms of classification accuracy (in percent). Results in brackets are the improvements of DAGNN over GCN.

#Training nodes per class	1	2	3	4	5	10	20	30	40	50	100
MLP	30.3	35.0	38.3	40.8	44.7	53.0	59.8	63.0	64.8	65.4	64.0
GCN	34.7	48.9	56.8	62.5	65.3	74.3	79.1	80.8	82.2	82.9	84.7
GAT	45.3	58.8	66.6	68.4	70.7	77.0	80.8	82.6	83.4	84.0	86.1
APPNP	44.7	58.7	66.3	71.2	74.1	79.0	81.9	83.2	83.8	84.3	85.4
SGC	43.7	59.2	67.2	70.4	71.5	77.5	80.4	81.3	81.9	82.1	83.6
DAGNN (Ours)	**58.4**(23.7↑)	**67.7**(18.8↑)	**72.4**(15.6↑)	**75.5**(13.0↑)	**76.7**(11.4↑)	**80.8**(6.5↑)	**83.7**(4.6↑)	**84.5**(3.7↑)	**85.6**(3.4↑)	**86.0**(3.1↑)	**87.1**(2.4↑)

由于接收场的扩大，对于很少的训练数据情况下，可以比其他的网络有巨大的优势

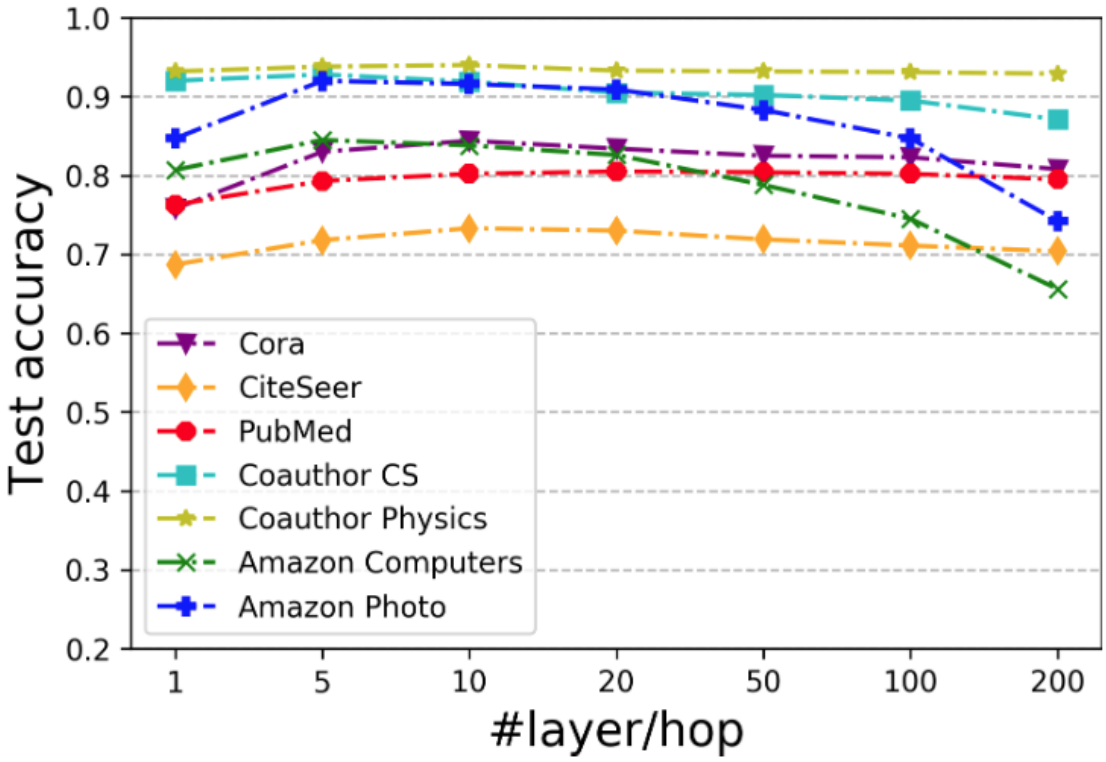


Figure 6: Results of DAGNN with different depths.

对于边密集的数据集，在多次迭代后，还是会出现过度平滑的问题，因为同一个连通块的值最后会趋于相同。

Conclusion

在本文中，考虑了当前深图神经网络中存在的性能下降问题，并针对深图神经网络发展了新的见解。

先对此问题进行系统的分析，认为损害网络性能的关键因素是特征转换和传播的纠缠。

提出建议对这两个操作进行解耦，并表明没有这种纠缠的深度图神经网络可以利用较大的接受域而不会导致性能下降。

提出DAGNN进行节点表示学习，并具有从大型自适应适应场中捕获信息的能力，DAGNN的性能要比当前最先进的模块好得多，尤其是在训练样本有限的情况下，这证明了它的优越性