Training Graph Neural Networks with 1000 Layers

https://arxiv.org/abs/2106.07476

研究背景：

在推荐系统中，我们的每一条query背后都是一张巨大的图。电影、商品、网站搜索都有这一张巨大的图网络，含有数十亿的节点和边。过去的图神经网络已经在小规模的图任务上取得了成功，但显存成为了将这些模型应用到大规模图的一个主要瓶颈。一些简单的方法有减少参数量，但这种方式有损于模型性能。图神经网络同时显示出其在大规模图上的潜能。其中DeeperGCN和SGC使用了很大的参数量并取得了很好的效果。

同时，有一些学者尝试去客服图神经网络对存储要求高的瓶颈。GraphSAGE和SGC在每次更新节点时只使用一部分节点计算梯度。而Cluster-GCN和GraphSAINT将一些节点聚合成为大一点的节点，减少了节点和边的数量。这些方式有两个问题：一是需要调试新的超参，二是如果将图分割而成的子图过小，很有可能破坏了原有的图结构。并且随着模型层数加深，这些方式仍然需要很大量的显存。

还有一种方式是diffusion. 这个我没怎么看过paper，之后会补上。这里贴上原文： However, the propagation schemes of these methods are non-trainable, which may lead to sub-optimality.

就是说这种模型的传播方式没法训练

模型

在搭建深度图神经网络时，作者在模型中采用了两种方法

1）残差连接

2）分组逆向图神经网络（Grouped Reversible GNNs）简写为逆向GNN

首先，将输入的特征矩阵 $X \in \mathbb{R}^{N*D}$ 平均分割为C组。每组的矩阵为 $X_i \in \mathbb{R}^{N*\frac{D}{C}}$ 。一个逆向GNN将分组后的 $\langle X_1, X_2, ..., X_C \rangle$ 映射为 $\langle X'_1, X'_2, ..., X'_C \rangle$ 。定义逆向GNN的**前向传播**为

$$X'_0 = \sum_{i=2}^{C} X_i$$

$$X'_i = f_{wi}(X'_{i-1}, A, U) + X_i, i \in (1, \cdots, C)$$

为了说明为什么逆向GNN可以节约存储，我们假设只有两组。

GNN第一组，第二组

$$X'_1 = f_{wi}(X_1, A, U) + X_1$$

$$X'_2 = f_{wi}(X_2, A, U) + X_2$$

RevGNN第一组，第二组

$$X_1' = f_{wi}(\sum_{i=2}^{C} X_i, A, U) + X_1$$

$$X_2' = f_{wi}(X_1', A, U) + X_2$$

这里我们就能看出，前者反向传播时由 X_2' 是无法推导出 X_2 的。而使用RevGNN后，反向传播时由 $X_2 = X_2' - f_{wi}(X_1', A, U)$ 就能推导出 X_2 。

我们已经有了 $X_2 ... X_C$ ，将他们相加就是 X_0' 。这样就有了 X_1 的推导公式：

$$X_1 = X_1' - f_{wi}(X_0', A, U)$$

这样，ResGNN不需要存储所有隐藏层的输出，只需要存储最后一层的就能反向传播出所有层。这样将原来需要存储为O(NDL)缩短为O(ND)。

3）减少参数量：参数共享的图神经网络

作者尝试了将每一层的参数设为同一套，应用在ResGNN和RevGNN中。

4）Deep Equilibrium GNN

这种方式假设模型会收敛到x = f(x)，然后用求根的方式去求解。

实验结果

作者进行了大量的实验

图1、2：逆向GNN只需要用很小量的存储就能超越ogbn-protein数据集的SOTA

Table 1. **Results on the *ogbn-proteins* dataset compared to SOTA.** RevGNN-Deep has 1001 layers with 80 channels each. It achieves SOTA performance with minimal GPU memory for training. RevGNN-Wide has 448 layers with 224 channels each. It achieves the best accuracy while consuming a moderate amount of GPU memory.

Model	ROC-AUC ↑	Mem ↓	Params
GCN (Kipf & Welling)	72.51 ± 0.35	4.68	96.9k
GraphSAGE (Hamilton et al.)	77.68 ± 0.20	3.12	193k
DeeperGCN (Li et al.)	86.16 ± 0.16	27.1	2.37M
UniMP (Shi et al.)	86.42 ± 0.08	27.2	1.91M
GAT (Veličković et al.)	86.82 ± 0.21	6.74	2.48M
UniMP+CEF (Shi et al.)	86.91 ± 0.18	27.2	1.96M
Ours (RevGNN-Deep)	87.74 ± 0.13	**2.86**	20.03M
Ours (RevGNN-Wide)	**88.24** ± 0.15	7.91	68.47M

Table 2. **Results on the *ogbn-arxiv* dataset compared to SOTA.** RevGCN-Deep has 28 layers with 128 channels each. It achieves SOTA performance with minimal GPU memory. RevGAT-Wide has 5 layers with 1068 channels each. RevGAT-SelfKD denotes the student models with 5 layers and 768 channels. It achieves the best accuracy while consuming a moderate amount of GPU memory.

Model	ACC ↑	Mem ↓	Params
GraphSAGE (Hamilton et al.)	71.49 ± 0.27	1.99	219k
GCN (Kipf & Welling)	71.74 ± 0.29	1.90	143k
DAGNN (Liu et al.)	72.09 ± 0.25	2.40	43.9k
DeeperGCN (Li et al.)	72.32 ± 0.27	21.6	491k
GCNII (Chen et al.)	72.74 ± 0.16	17.0	2.15M
GAT (Veličković et al.)	73.91 ± 0.12	5.52	1.44M
UniMP_v2 (Shi et al.)	73.97 ± 0.15	25.0	687k
Ours (RevGCN-Deep)	73.01 ± 0.31	**1.84**	262k
Ours (RevGAT-Wide)	74.05 ± 0.11	8.49	3.88M
Ours (RevGAT-SelfKD)	**74.26** ± 0.17	6.60	2.10M

图3：不同的图卷积操作对结果的影响。可以看出GAT最好，但是参数量也最多。

Table 3. **Results with different GNN operators on the *ogbn-arxiv*.** All GAT models use label propagation. #L and #Ch denote the number of layers and channels respectively. *Baselines* are in italic.

Model	#L	#Ch	ACC ↑	Mem ↓	Params
ResGCN	28	128	72.46 ± 0.29	11.15	491k
RevGCN	28	128	73.01 ± 0.31	**1.84**	262k
RevGCN	28	180	**73.22** ± 0.19	2.73	500k
ResSAGE	28	128	72.46 ± 0.29	8.93	950k
RevSAGE	28	128	72.69 ± 0.23	**1.17**	491k
RevSAGE	28	180	**72.73** ± 0.10	1.57	953k
ResGEN	28	128	72.32 ± 0.27	21.63	491k
RevGEN	28	128	72.34 ± 0.18	**4.08**	262k
RevGEN	28	180	**72.93** ± 0.10	5.67	500k
ResGAT	5	768	73.76 ± 0.13	9.96	3.87M
RevGAT	5	768	74.02 ± 0.18	**6.30**	2.10M
RevGAT	5	1068	**74.05** ± 0.11	8.49	3.88M

一些腐化实验（Ablation Studies）

- 在ogbn-products数据集中，RevGNN表现明显优于SGC和SIGN。
- 在Ogbg-molhiv数据集中，同样使用14层256隐层大小的RevGNN，GraphNorm稍微优于BatchNorm。
- 在ogbn-products数据集中，Mini-batch training将准确率从full-batch的78.77%提升到82.16%，而仅仅使用了后者44%的内存。