
图神经网络的回顾

A PREPRINT

袁铭潮

December 13, 2019

1 背景

随着机器学习和深度学习的发展，越来越多的任务与深度学习开始结合，同时，现有的技术的局限性也显得更为突出。许多学习任务需要处理图形数据，图形数据包含元素之间的丰富关系信息。建模物理系统，学习分子指纹，预测蛋白质界面以及对疾病进行分类都需要一个模型来从图形输入中学习。

在诸如从文本和图像之类的非结构数据中学习的其他领域中，对提取的结构的推理，例如句子的依存关系树和图像的场景图，是一个重要的研究课题，它也需要图推理模型。

由于图的强大表达能力，利用机器学习来分析图的研究受到了越来越多的关注。近年来，基于图神经网络变体的系统，例如图神经网络（GNN），图卷积网络（GCN），门控图神经网络（GGNN），图注意力网络（GAT），在上述许多任务上均展现了突破性的性能。

这里根据回顾之前阅读的图神经网络相关的文章，进行简单的梳理总结。

2 研究现状

首先，对近年来的图神经网络方面的发展进行回顾：图神经网络的概念最早在2005年提出。2009年Franco博士在其论文中定义了图神经网络的理论基础。

最早的GNN主要解决的还是如分子结构分类等严格意义上的图论问题。但实际上欧式空间(比如像图像Image)或者是序列(比如像文本Text)，许多常见场景也都可以转换成图(Graph)，然后就能使用图神经网络技术来建模。2009年后图神经网络也陆续有一些相关研究。

直到2013年，在图信号处理(Graph Signal Processing)的基础上，Bruna首次提出图上的基于频域(Spectral-domain)和基于空域(Spatial-domain)的卷积神经网络。其后至今，学界提出了很多基于空域的图卷积方式，也有不少学者试图通过统一的框架将前人的工作统一起来。而基于频域的工作相对较少，只受到部分学者的青睐。

计划将图卷积网络部分的内容另外进行综述，本文内容涉及图卷积网络提出前的各类网络及模型主要包含GNN和GGNN两篇论文。

3 图神经网络GNN

图神经网络起源于Franco博士的论文[1]，它的理论基础是不动点理论。它扩展了现有的神经网络，用于处理图域中表示的数据。在图中，每个节点自然是由其特征和相关节点定义的。

GNN的学习目标是获得每个结点的图感知的隐藏状态 h_v (state embedding)，这就意味着：对于每个节点，它的隐藏状态包含了来自邻居节点的信息。GNN通过迭代式更新所有结点的隐藏状态来实现每个结点感知到图上的其他结点。GNN通过迭代式更新所有结点的隐藏状态来实现，在 $t+1$ 时刻，结点 v 的隐藏状态按照如下方式更新：

$$h_v^{(t)} = f^*(l_v, l_{CO(v)}, l_{NBR(v)}, h_{NBR(v)}^{(t-1)}) \quad (1)$$

其中 l_v 表示节点标签: $l_v \in \{1, \dots, L_V\}$, l_e 表示边标签: $l_e \in \{1, \dots, L_E\}$, 前导节点集合: $IN(v) = \{v' | (v', v) \in \mathcal{E}\}$, 后继节点集合: $OUT(v) = \{v' | (v, v') \in \mathcal{E}\}$, 邻节点集合: $NBR(v) = IN(v) \cup OUT(v)$, 节点的边集合: $CO(v) = \{(v', v'') \in \mathcal{E} | v = v' \vee v = v''\}$ 。

这里需要介绍一下不动点理论, GNN所使用的不动点理论是这里的不动点理论专指巴拿赫不动点定理(Banach's Fixed Point Theorem)。上式的更新公式可以写成:

$$h_v^{(t)} = f(h_v^{(t-1)}, X) \quad (2)$$

不论 h 是什么, 只要 f 是个压缩映射(contraction map), h 经过不断迭代都会收敛到某一个固定的点, 我们称之为不动点。

GNN的Loss函数是根据不动点原理的限制所得到的, 这里就不赘述了。GNN中压缩映射函数通过一个简单的前向神经网络即可实现。

因为理论基础是不动点理论, 所以GNN是通过Almeida-Pineda算法完成收敛的, 优点是不需要存储中间状态进行梯度计算, 缺点则是必须限制参数, 传播步骤是收缩图, 这是确保收敛的必要条件, 但可能会限制模型的表达。

其实GNN的模型与已有的RNN的模型十分的近似, 如图:

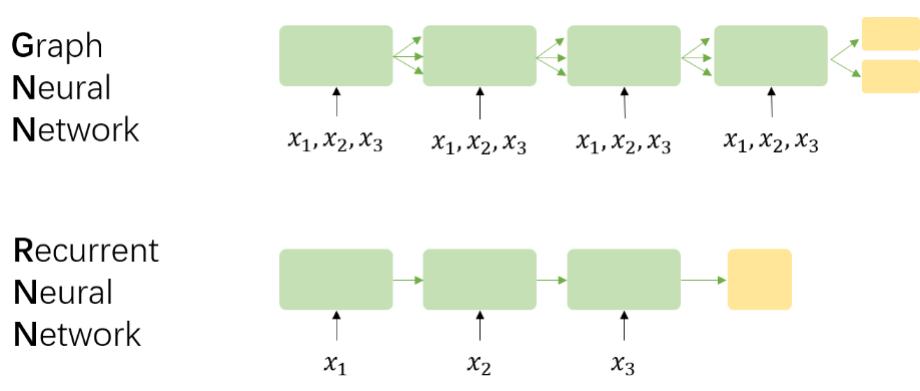


Figure 1: GNN与RNN对比

可以看出两个在形态上十分相似, 但仍存在较大不同:

- 1、GNN基于不动点理论, 展开长度根据收敛条件确定;RNN的展开长度为序列本身长度。
- 2、GNN采用AP算法, 对收敛性有要求;RNN采用BP算法, 对收敛性无要求。
- 3、GNN的输入是所有节点的特征, 时间步之间的信息流由边决定;RNN的输入是该时刻对应的输入, 时间步由读入顺序决定。
- 4、GNN只有在隐藏状态收敛后才能输出;RNN在每个时间步上都可以输出。

4 门控图神经网络GGNN

GNN的提出以及其与RNN相似的形态, 以及许多任务对过程中学习内部状态表示获取的需求, 多伦多大学的一个团队提出了GGNN的概念及模型[2]。

由于GNN收敛性的限制, GGNN考虑不以不动点理论为基础对GNN进行修改, 这也就意味着可以使用BP算法来进行训练, 迭代不需要收敛才输出。

如图1所示, GNN与RNN形态上相近, 也就很容易能套用RNN的公式来实现, 论文中指出, 他们首先是考虑的RNN模型, 但是效果并不理想, 于是使用了与比RNN更优秀的GRU模型, 虽然形态上并不直观, 总体的思路是不变的, 就是采两者所长, 得到一个较两者更好的模型。

与GNN定义的范式一致, GGNN也有两个过程: 状态更新与输出。相比GNN而言, 它主要的区别来源于状态更新阶段。具体地, GGNN参考了GRU的设计, 把邻居结点的信息视作输入, 结点本身的状态视作隐藏状态。

GNN中由于收缩映射约束确保了固定点与初始化无关，GGNN使用的并不是收缩映射的方式，所以还是需要进行初始化的，于是可以将节点标签合并为其他输入，增加了向量 x 表示，并为了和标签区别，称为节点注释。传播模型很容易学习将节点注释传播到所有可到达的节点。

GGNN的本质是将GNN模型与GRU进行结合，得到更普适的图模型，减少了GNN模型的限制，总体来看甚至与RNN的限制是相差无几的，这也使得GGNN能够更广泛的使用到其他领域中。结合了GRU的GGNN不仅能够适用于图数据，同时还能够处理序列数据。

$$\begin{aligned}
 h_v^{(1)} &= [x_v^\top, 0]^\top \\
 a_v^{(t)} &= A_{v:}^\top [h_1^{(t-1)\top} \dots h_{|\mathcal{V}|}^{(t-1)\top}]^\top + b \\
 r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\
 z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\
 \tilde{h}_t &= \tanh(W_{\tilde{h}} \cdot [r_t * h_{t-1}, x_t]) \\
 h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \\
 y_t &= \sigma(W_o \cdot h_t) \\
 h_v^{(t)} &= \widetilde{h}_v^{(t)} = \tanh(Wa_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \\
 h_v^{(t)} &= (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \widetilde{h}_v^{(t)}
 \end{aligned}$$

Figure 2: GRU与GGNN公式对比

简单的对GRU和GGNN的公式进行两对比，GGNN在GRU的基础上，在 r_v^t 、 h_v^t 、 z_v^t 的输入中增加了 $a_v^{(t)}$ 。其中矩阵 $A \in D|\mathcal{V}| \times 2D|\mathcal{V}|$ 等价于 $A = [A^{(out)}, A^{(in)}]$ ，上式中的 $A_{v:} \in D|\mathcal{V}| \times 2D$ ，于是 $a_v^{(t)} \in 2D$ 。

GGNN目前得到了广泛的应用，相比于GNN，其最大的区别在于不再以不动点理论为基础，虽然这意味着不再需要迭代收敛，但同时它也意味着GGNN的初始化很重要。

从另一个角度来看，虽然GNN与GGNN的理论不同，但从设计哲学上来看，它们都与循环神经网络的设计类似。

循环神经网络的好处在于能够处理任意长的序列，但它的计算必须是串行计算若干个时间步，时间开销不可忽略。所以，上面两种基于循环的图神经网络在更新隐藏状态时不太高效。如果借鉴深度学习中堆叠多层的成功经验，我们有足够的理由相信，多层图神经网络能达到同样的效果。基于循环的图神经网络每次迭代时都共享同样的参数，而多层神经网络每一层的参数不同，可以看成是一个层次化特征抽取(Hierarchical Feature Extraction)的方法。

5 借鉴与意义

关于GCN、GAT的论文会在后续进行梳理，目前来看GCN已经开始被广泛的使用到各领域中了，重要性不言而喻。

GNN及其变形的论文的应用确并不太广泛，可能是计算量的问题，也可能是数据集的问题，同样可能是GNN尚不能构建深层网络的问题。这个需要后续在文章的阅读中找到答案。

重新梳理较为经典的模型与论文，让我能对其有更深入的理解，本来一知半解的点，在重新学习梳理后理解更加深刻了，一些以前没有完全明白的内容也理解了很多。当然，还是有部分没有完全理解，不过对于目前来说已经足够了。

之后会尝试以复现的方式来对经典的论文模型进行学习，以求达到学以致用的目的。

References

- [1] Scarselli F , Gori M , Tsoi A C , et al. The Graph Neural Network Model[J]. IEEE Transactions on Neural Networks, 2009, 20(1):61-80.
- [2] Li Y , Tarlow D , Brockschmidt M , et al. Gated Graph Sequence Neural Networks[J]. Computer Science, 2015.b2e000a.