《Graph Attention Networks》

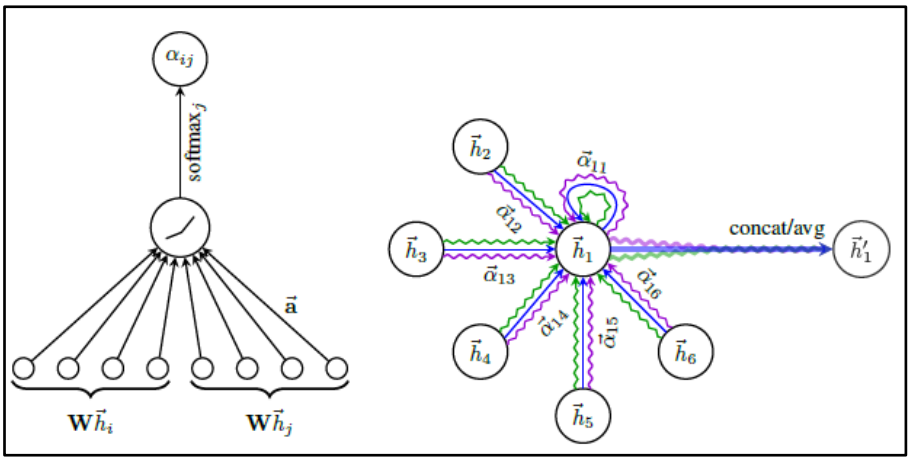
《图注意力网络》

摘要：

1. **本文的背景：**GCN 的提出虽然给出了一个图学习问题上的有效解决方法，但其本身却存在许多缺陷。针对这些设计上的缺陷，众多研究者们也纷纷开始提出了各自的改进方案，例如 GraphSAGE，而本文是一种基于注意力机制的解决方案。
2. **本文的贡献：**提出了一种新型的神经网络架构——**图注意力网络（GAT）**。不同于 GCN，GAT 能够通过 attention 层给每个邻居节点分配不同的权重，从而能够识别出更加重要的邻居。此外，GAT 还是一种**归纳式学习**。
3. **主要创新点：**通过采用多头注意力架构来稳定学习邻居节点集对目标节点的权重分配。并且，上述这种方式可以直接应用到归纳式学习问题中。
4. **实验结果：**GAT 模型在四个已公开的直推式和归纳式图数据集上取得了与当时的最优水平相当或更好的结果，进而证明了 GAT 模型可以有效地适用于（基于图的）归纳学习问题与直推学习问题。

图注意力层：

一个图注意力层（Graph Attentional Layer）的结构如下图所示：



它的输入是一组节点特征，记为：

$$h = \{\vec{h}_1, \vec{h}_2, \vec{h}_3, \dots, \vec{h}_N\}, \qquad \vec{h}_i \in \mathbb{R}^F$$

其中，N是节点的个数，F是每个节点的特征数（维度）。

而输出是一组新的节点特征，记为：

$$h' = \left\{\vec{h}_1{}', \vec{h}_2{}', \vec{h}_3{}', \dots, \vec{h}_N{}'\right\}, \qquad \vec{h}_i{}' \in \mathbb{R}^{F'}$$

其中，节点个数N不变，而每个节点的特征数（维度）可变化为F'。

（1） 在开始计算 attention 之前，首先会对所有节点做一次共享的线性变换以获得特征增强，也就是将输入特征转换为高维特征，即：

$$\vec{h}_i \Longrightarrow W\vec{h}_i$$

其中，$W \in \mathbb{R}^{F' \times F}$是一个权重矩阵（被所有的$\vec{h}_i$共享）。

（2） 在所有节点上共享 self-attention 机制，计算节点i和j之间的 attention 系数。该系数表示了节点i的特征对节点j的重要性，即：

$$e_{ij} = a(W\vec{h}_i, W\vec{h}_j)$$

其中，$a(\cdot)$是一个$\mathbb{R}^{F'} \times \mathbb{R}^{F'} \longrightarrow \mathbb{R}$的映射。

（3） 一般来说，由于 self-attention 会将注意力分配到图中所有的节点上，而这种做法显然会丢失结构信息。所以，为了解决这一问题，本文使用了一种 masked attention 的方式——仅将注意力分配到节点i的邻节点集上，即$j \in \mathcal{N}_i$，然后再进行归一化处理：

$$\alpha_{ij} = softmax_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})}$$

（4） 因此，单层 Layer 的总的计算过程为：

$$\alpha_{ij} = \frac{\exp\left(LeakyReLU\left(\vec{a}^T[W\vec{h}_i || W\vec{h}_j]\right)\right)}{\sum_{k \in \mathcal{N}_i} \exp\left(LeakyReLU\left(\vec{a}^T[W\vec{h}_i || W\vec{h}_k]\right)\right)}$$

其中，$||$表示拼接，$\vec{a}^T \in \mathbb{R}^{2F'}$为前馈神经网络$a(\cdot)$的参数，$LeakyReLU$为前馈神经网络的激活函数。

（5） 于是，我们就可以得到$\vec{h}_i{}'$为：

$$\vec{h}_i{}' = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} W\vec{h}_j\right)$$

其中，σ表示激活函数。并且注意：$\mathcal{N}_i$中包含节点i本身。

（6） 而为了提高模型的拟合能力，在本文中作者还引入了多头 self-attention 机

制，即同时使用多个W^k计算 self-attention，然后将各个W^k计算得到的结果合并（连接或求和），即：

$$\vec{h}_i{}' = ||_{k=1}^{K}\sigma\left(\sum_{j\in\mathcal{N}_i}\alpha_{ij}^k W^k\vec{h}_j^k\right) \qquad (1)$$

$$\vec{h}_i{}' = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in\mathcal{N}_i}\alpha_{ij}^k W^k\vec{h}_j^k\right) \qquad (1)$$

其中，||表示拼接，α_{ij}^k和W^k表示第k个多头得到的计算结果。而由于$W^k \in \mathbb{R}^{F'\times F}$，因此在(1)中的$\vec{h}_i{}' \in \mathbb{R}^{KF'}$，而(2)中的$\vec{h}_i{}' \in \mathbb{R}^{F'}$。

模型比较：

GCN 的消息传递机制版本：

$$h_i^{(l+1)} = \sigma\left(b^{(l)} + \sum_{j\in\mathcal{N}_i}\frac{1}{c_{ji}}h_j^{(l)}W^{(l)}\right)$$

$$c_{ji} = \sqrt{|\mathcal{N}_j|}\sqrt{|\mathcal{N}_i|}$$

GCN 的矩阵版本：

$$H^{(l+1)} = \hat{A}H^{(l)}W^{(l)}$$

$$\hat{A} = \hat{D}^{-\frac{1}{2}}\tilde{A}\hat{D}^{-\frac{1}{2}}$$

GAT 的消息传递机制版本：

$$h_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}_i}\alpha_{ij}^{(l)}W^{(l)}h_j^{(l)}\right)$$

$$\alpha_{ij}^{(l)} = softmax_j\left(e_{ij}^{(l)}\right)$$

$$e_{ij}^{(l)} = LeakyReLU\left(\vec{a}^{T(l)}[W^{(l)}h_i^{(l)}||W^{(l)}h_j^{(l)}]\right)$$

由此可见，GCN 的“注意力系数”$\frac{1}{c_{ij}}$是固定的，而 GAT 的注意力系数α_{ij}是能够自适应的。

Graph Attention 机制的特性：

(1) 操作效率高，在跨节点对中可并行计算；

(2) 通过对相邻节点指定任意权值，可应用于不同度的图节点；

(3) 可直接适用于归纳式学习问题，包括将模型推广到完全不可见图的任务。

实验及结果：

（1）数据集

本文的实验建立在四个基于图的任务上，这些任务包括三个直推式学习（transductive learning）任务以及一个归纳式学习（inductive learning）任务。具体数据集情况如下：

Table 1: Summary of the datasets used in our experiments.

	Cora	Citeseer	Pubmed	PPI
Task	Transductive	Transductive	Transductive	Inductive
# Nodes	2708 (1 graph)	3327 (1 graph)	19717 (1 graph)	56944 (24 graphs)
# Edges	5429	4732	44338	818716
# Features/Node	1433	3703	500	50
# Classes	7	6	3	121 (multilabel)
# Training Nodes	140	120	60	44906 (20 graphs)
# Validation Nodes	500	500	500	6514 (2 graphs)
# Test Nodes	1000	1000	1000	5524 (2 graphs)

其中，Cora、Citeseer 与 Pubmed 数据集用于 Transductive Learning，而 PPI 数据集用于 Inductive Learning。

（2）直推式学习任务及结果

在直推式学习任务中，实验采用了**双层 GAT 模型**。第一层由$K = 8$个注意力头组成，每个注意力头计算$F' = 8$个特征（共64个特征），然后通过指数线性单元（ELU）进行非线性激活。第二层用于分类：单个注意力头计算C个特征（其中C为类别数），然后进行$softmax$激活。此外，为了应对较小的训练集，实验在模型中大量使用了**正则化**——在训练过程中，采用$L2$正则化，$\lambda = 0.0005$；而两层输入以及归一化注意力系数都采用了$p = 0.6$的$dropout$。

因此，最终的实验结果如下表所示：

Table 2: Summary of results in terms of classification accuracies, for Cora, Citeseer and Pubmed. GCN-64* corresponds to the best GCN result computing 64 hidden features (using ReLU or ELU).

Transductive

Method	Cora	Citeseer	Pubmed
MLP	55.1%	46.5%	71.4%
ManiReg (Belkin et al., 2006)	59.5%	60.1%	70.7%
SemiEmb (Weston et al., 2012)	59.0%	59.6%	71.7%
LP (Zhu et al., 2003)	68.0%	45.3%	63.0%
DeepWalk (Perozzi et al., 2014)	67.2%	43.2%	65.3%
ICA (Lu & Getoor, 2003)	75.1%	69.1%	73.9%
Planetoid (Yang et al., 2016)	75.7%	64.7%	77.2%
Chebyshev (Defferrard et al., 2016)	81.2%	69.8%	74.4%
GCN (Kipf & Welling, 2017)	81.5%	70.3%	**79.0%**
MoNet (Monti et al., 2016)	81.7 ± 0.5%	—	78.8 ± 0.3%
GCN-64*	81.4 ± 0.5%	70.9 ± 0.5%	**79.0** ± 0.3%
GAT (ours)	**83.0** ± 0.7%	**72.5** ± 0.7%	**79.0** ± 0.3%

可以看到，GAT 模型的效果要基本优于当时的其他模型。

（3）归纳式学习任务及结果

在归纳式学习任务中，实验采用了**三层*GAT*模型**。前两层都由$K = 4$个注意力头组成，计算$F' = 256$个特征（共1024个特征），然后是ELU非线性。最后一层用于（多标签）分类：$K = 6$个注意头，每个注意头计算121个特征，取平均值，然后进行 sigmoid 激活。此外，由于这项任务的训练集足够大，所以没有必要应用$L2$正则化或$dropout$。

因此，最终的实验结果如下表所示：

Table 3: Summary of results in terms of micro-averaged F_1 scores, for the PPI dataset. GraphSAGE* corresponds to the best GraphSAGE result we were able to obtain by just modifying its architecture. Const-GAT corresponds to a model with the same architecture as GAT, but with a constant attention mechanism (assigning same importance to each neighbor; GCN-like inductive operator).

Inductive

Method	PPI
Random	0.396
MLP	0.422
GraphSAGE-GCN (Hamilton et al., 2017)	0.500
GraphSAGE-mean (Hamilton et al., 2017)	0.598
GraphSAGE-LSTM (Hamilton et al., 2017)	0.612
GraphSAGE-pool (Hamilton et al., 2017)	0.600
GraphSAGE*	0.768
Const-GAT (ours)	0.934 ± 0.006
GAT (ours)	**0.973** ± 0.002

可以看到，GAT 模型的效果要远远优于当时的其他模型。

总结：

本文提出了一种基于$self-attention$的新型图神经网络架构——GAT。总的来说，GAT 的特点主要有以下两点：

（1） 在 GAT 中，图中的每个节点可以根据邻居节点的特征而为其分配不同的权值，并且这个权重是可学习的。

（2） 在引入自注意力机制之后，GAT 的目标节点计算只与邻居节点有关，而无需得到整张图的信息。因此，GAT 模型不仅可扩展到有向图的问题中，还可以直接适用于 Inductive Learning——包括在训练期间完全看不见的图上的评估模型的任务。

对本文的感悟：

与 GraphSAGE 类似，GAT 也是在针对 GCN 一系列的缺陷上所提出的改进方案。不过，它有效结合了当时来自 NLP 领域上 Transformer 模型所提出的自注意力（$self-attention$）概念。也正是这一特点，使得它的效果远优于当时的其他模型。而这种来自其他领域的相互借鉴也在之前的“DeepWalk”论文中类似出现过，而这正是我们在学术研究中最值得借鉴和关注的地方，即如何交叉学科应用。