

# 《Graph Transformer Networks》

## 《图转换网络》

### 摘要:

1. **本文的背景:** 当前,大多数 GNN 存在一个局限性:只能处理同构图。然而,生活中大多数应用场景都是异构的,传统的 GNN 并不能处理这种关系,虽然有些做法是忽略这些节点和边的类型,将它们看成是同构图来进行处理,但这会丧失了原图中的一些信息。此外,现有的一些处理异构图的方法是采用**元路径**,即手工将异构图转化为由元路径定义的同构图,然后再使用传统的 GNN 方法进行处理,但这种手工设计元路径的方式费时又费力。因此,为了解决上述问题,本文作者提出了一种以端到端的方式学习异构图上节点表示的新型框架。
2. **本文的贡献:** 提出了一种新型的图神经网络框架 GTN——Graph Transformer Network,通过识别有用的元路径和多跳连接来学习图上的有效节点表示;表明图的生成是可解释的,GTN 模型能够提供用于预测的有效元路径。
3. **主要创新点:** 采用了端到端的方式对异构图问题进行处理,避免了人工设计;利用了多卷积通道的方式并行考虑了多种类型的元路径。
4. **实验结果:** 实验有力证明了 GTN 模型学习到的节点表示的有效性,在与异构图的所有三个基准节点数据分类问题中额外使用领域知识的最新方法相比,GTN 取得了最佳性能。

### 同构图与异构图:

同构图:仅具有一种类型的节点和边的标准图。

异构图:具有多种类型的节点和边的标准图。例如,引用网络具有多种类型的节点(如作者、论文、会议)和由它们之间的关系定义的边(如作者—论文、论文—会议)。

### 相关定义:

1. 图 $G$ 表示为 $G = (V, E)$ ,其中 $V$ 代表节点的集合, $E$ 代表可观测的边的集合。
2.  $\mathcal{T}^v$ 和 $\mathcal{T}^e$ 分别表示节点的种类和边的种类集合。

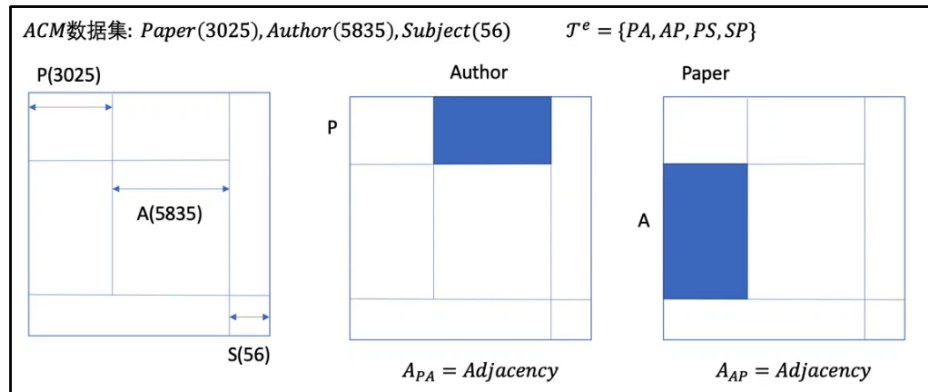
3. 异构图表示为邻接矩阵的集合 $\mathbb{A} = \{A_k\}_{k=1}^K$ ，其中 $K = |\mathcal{T}^e|$ ， $A_k \in \mathbb{R}^{N \times N}$ 代表其中一个邻接矩阵（ $N$ 为节点的数目），而 $A_k[i, j] \neq 0$ 则代表节点 $i$ 到节点 $j$ 存在第 $k$ 类型的边。此外， $\mathbb{A}$ 也可以写成张量形式 $\mathbb{A} \in \mathbb{R}^{N \times N \times K}$ 。
4. 节点的特征矩阵定义为 $X \in \mathbb{R}^{N \times d}$ ，其中 $d$ 表示节点特征的维度。
5. 元路径用 $p$ 来表示，它在异构图 $G$ 上由一些异构边连接而成，即：

$$v_1 \xrightarrow{t_1} v_2 \xrightarrow{t_2} \dots v_l \xrightarrow{t_l} v_{l+1}$$

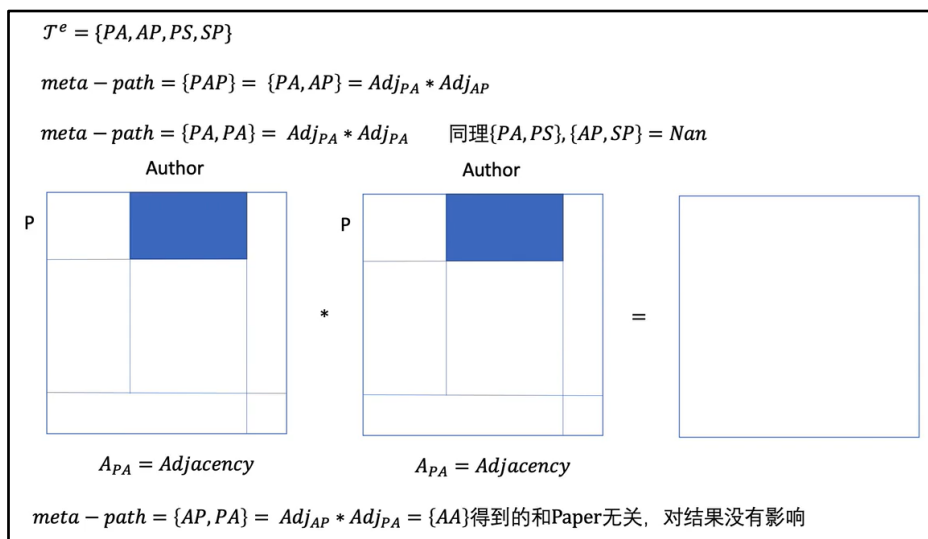
其中， $t_l \in \mathcal{T}^e$ 表示第 $l$ 条边的类型。

6. 元路径 $P$ （复合关系为 $R = t_1 \circ t_2 \dots \circ t_l$ ）的邻接矩阵 $A_P$ 可通过邻接矩阵的乘法来得到：

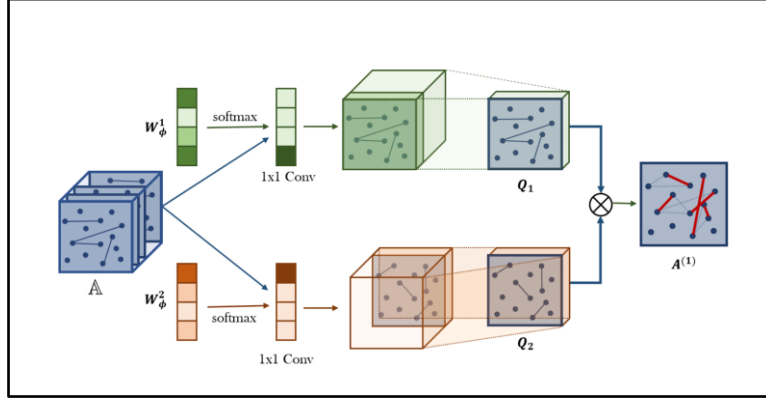
$$A_P = A_{t_1} A_{t_2} \dots A_{t_l}$$



如何组合多元的元路径：



元路径的生成实现：



上图表示的是 GT(Graph Transformer)层，它首先从张量 $\mathbb{A}$ （每一片代表一种边类型）中用权重参数来重新选择邻接矩阵（即边类型），而权重参数分配的方式可以理解成对 $\mathbb{A}$ 进行卷积。注意，卷积后的两个矩阵是两个不同的图结构，表示为 $Q_1$ 和 $Q_2$ 。然后，我们再将这两个矩阵组合成新的图结构（即运用矩阵乘法）：

$$A^{(1)} = Q_1 Q_2$$

如果用具体的数学公式进行表示，则表达如下：

1) 选择的 $Q$ 可以表示为：

$$Q = F(\mathbb{A}; W_\phi) = \phi(A; \text{softmax}(W_\phi))$$

其中， $\mathbb{A}$ 表示异构图邻接矩阵的集合， $W_\phi$ 表示权重参数。

2) 每一个 $Q_i$ 可进一步表示成：

$$Q_i = \sum_{t_l \in \mathcal{T}^e} \alpha_{t_l}^{(l)} A_{t_l}$$

其中， $\mathcal{T}^e$ 是边的类型集合， $\alpha_{t_l}^{(l)}$ 是边第 $l$ 种类型 $t_l$ 在第 $l$ 层的权重。

3) 举例，如上图所示：

假设 $\mathcal{T}^e = \{t_1, t_2, t_3, t_4\}$ ，则 $\mathbb{A} = \{A_1, A_2, A_3, A_4\}$ 。若 $W_\phi^1 = \{\alpha_1^1, \alpha_2^1, \alpha_3^1, \alpha_4^1\}$ ， $W_\phi^2 = \{\alpha_1^2, \alpha_2^2, \alpha_3^2, \alpha_4^2\}$ （已经进行过 $\text{softmax}$ 归一化），则：

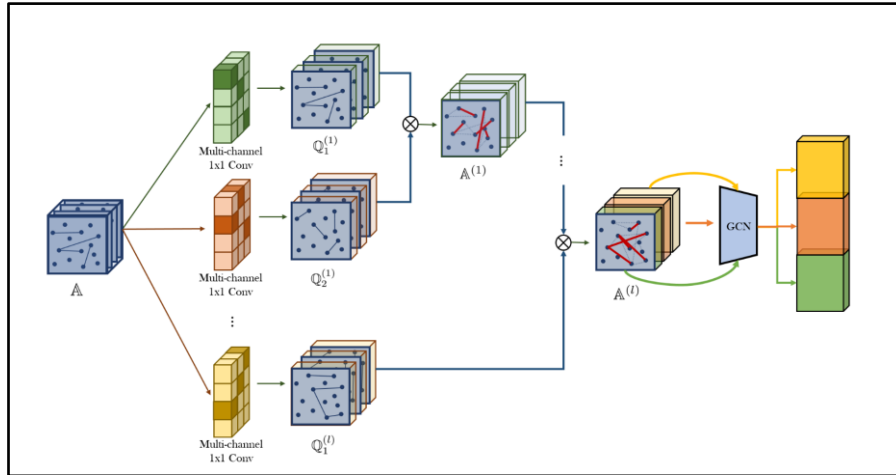
$$Q_1 = \alpha_1^1 * A_1 + \alpha_2^1 * A_2 + \alpha_3^1 * A_3 + \alpha_4^1 * A_4$$

$$Q_2 = \alpha_1^2 * A_1 + \alpha_2^2 * A_2 + \alpha_3^2 * A_3 + \alpha_4^2 * A_4$$

$$A^{(1)} = Q_1 Q_2 = \left( \sum_{t_1 \in \mathcal{T}^e} \alpha_{t_1}^{(1)} A_{t_1} \right) \left( \sum_{t_2 \in \mathcal{T}^e} \alpha_{t_2}^{(2)} A_{t_2} \right)$$

4) 如果不是仅分两个 $Q$ ，而是更多个，则最后得到的结果可表示为：

$$A_P = A^{(l)} = \left( \sum_{t_1 \in \mathcal{T}^e} \alpha_{t_1}^{(1)} A_{t_1} \right) \left( \sum_{t_2 \in \mathcal{T}^e} \alpha_{t_2}^{(2)} A_{t_2} \right) \dots \left( \sum_{t_l \in \mathcal{T}^e} \alpha_{t_l}^{(l)} A_{t_l} \right)$$



因此, $l$  个 GT 层的堆叠可以学习长度为 $l$ 的元路径图结构,如上图所示的 GTN 结构。但这种构造存在一个问题——添加 GT 层总是在增加元路径的长度,并不允许原始边的出现。而在某些应用中,长元路径和短元路径都很重要。

因此,为了学习包含原始边的短元路径和长元路径,我们将单位矩阵 $I$ 包含在了 $A$ 中,即 $A_0 = I$ 。更简单来说,在利用了这个技巧后我们能够使GTN学习到任意长度的元路径,且最高为 $l + 1$ ,即将 $l$ 个 GT 层进行堆叠。

### Graph Transformer Networks:

为了同时考虑多种类型的元路径,我们可以将上图中 $1 \times 1$ 卷积的输出通道设为 $C$ ,于是这些 GT 层就能产生一组元路径,即中间邻接矩阵 $Q_1$ 和 $Q_2$ 变成了邻接张量 $Q_1$ 和 $Q_2 \in \mathbb{R}^{N \times N \times C}$ 。(实验表明通过多个不同的图结构学习不同的节点表示是有益的。)

然后,在 $l$ 个 GT 层堆栈之后,我们再将GCN应用于元路径张量 $A^{(l)} \in \mathbb{R}^{N \times N \times C}$ 的每个通道上,并将多个节点的表示拼接起来。因此,最后生成的嵌入向量 $Z$ 可以表示为:

$$Z = ||_{i=1}^C \sigma \left( \tilde{D}_i^{-1} \tilde{A}_i^{(l)} X W \right)$$

其中, $||$ 表示拼接操作, $C$ 表示通道数量, $\tilde{A}_i^{(l)} = A_i^{(l)} + I$ 表示张量 $A^{(l)}$ 的第 $i$ 个通道, $\tilde{D}_i$ 是 $\tilde{A}_i$ 的度矩阵, $W \in \mathbb{R}^{d \times d}$ 是训练权重矩阵, $X \in \mathbb{R}^{N \times d}$ 表示特征矩阵。

## 实验及结果：

本论文通过实验和分析回答了以下研究问题：

Q1. GTN 生成的新图结构对学习节点表示是否有效？

Q2. GTN 能否根据数据集自适应地产生可变长度的元路径？

Q3. 如何从 GTN 生成的邻接矩阵来解释每个元路径的重要性？

## 数据集：

实验使用了两个文献网络数据集 DBLP 和 ACM，以及一个电影数据集 IMDB。而表 1 中显示了实验中使用的异构图的统计信息。

Table 1: Datasets for node classification on heterogeneous graphs.							
Dataset	# Nodes	# Edges	# Edge type	# Features	# Training	# Validation	# Test
DBLP	18405	67946	4	334	800	400	2857
ACM	8994	25922	4	1902	600	300	2125
IMDB	12772	37288	4	1256	300	300	2339

**说明：**DBLP 中包含三种类型的节点（论文 P，作者 A，会议 C），四种类型的边（PA、AP、PC、CP），并以作者的研究领域作为标签 label。ACM 中包含三种类型的节点（论文 P，作者 A，主题 S），四种类型的边（PA、AP、PS、SP），并以文章类别作为标签 label。而在 IMDB 中包含三种类型的节点（电影 M，演员 A 和导演 D），以电影的流派作为标签 label。

## 实现细节：

我们将所有方法的嵌入维度设置为 64，以便进行公平比较。使用 Adam 优化器并分别选择超参数（例如学习率，权重衰减等），以便每个基线方法产生其最佳性能。对于基于随机游走的模型，每个节点的游走长度设置为 100，并进行 1000 次迭代，窗口大小设置为 5，包含 7 个负样本。而对于 GCN、GAT 和 HAN，参数分别使用验证集进行优化。最后，对于本文的模型 GTN，我们对 DBLP 和 IMDB 数据集使用了三个 GT 层，对 ACM 数据集使用两个 GT 层。此外，我们用常量值对每个 GT 层中  $1 \times 1$  卷积的参数进行初始化。

节点分类结果：

Table 2: Evaluation results on the node classification task (F1 score).							
	DeepWalk	metapath2vec	GCN	GAT	HAN	GTN <sub>-I</sub>	GTN (proposed)
DBLP	63.18	85.53	87.30	93.71	92.83	93.91	<b>94.18</b>
ACM	67.42	87.61	91.60	92.33	90.96	91.13	<b>92.68</b>
IMDB	32.08	35.21	56.89	58.14	56.77	52.33	<b>60.92</b>

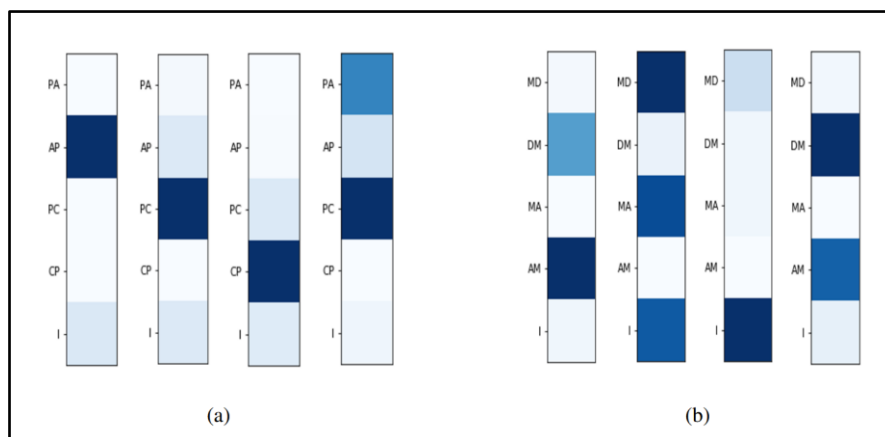
如表 2 所示，它显示了 GTN 和其他节点分类基线的性能。通过分析实验结果，我们能够回答问题 Q1 和 Q2。可以发现，相对于所有网络嵌入方法和图神经网络方法，GTN 模型在所有数据集上都实现了最好的性能。此外，GTN<sub>-I</sub> 表示候选邻接矩阵中没有  $I$ ，而从结果中我们可以看出其性能比包含  $I$  的要差，这证明了添加单位阵  $I$  的有效性。

可解释性：

作者经过公式推导得出，一条 meta-path（元路径） $t_l, t_{l-1}, \dots, t_1, t_0$  的贡献度能通过  $\prod_{i=0}^l \alpha_{t_i}^{(i)}$  进行获取。

Table 3: Comparison with predefined meta-paths and top-ranked meta-paths by GTNs. Our model found important meta-paths that are consistent with pre-defined meta-paths between target nodes (a type of nodes with labels for node classifications). Also, new relevant meta-paths between all types of nodes are discovered by GTNs.			
Dataset	Predefined Meta-path	Meta-path learnt by GTNs	
		Top 3 (between target nodes)	Top 3 (all)
DBLP	APCPA, APA	APCPA, APAPA, APA	CPCPA, APCPA, CP
ACM	PAP, PSP	PAP, PSP	APAP, APA, SPAP
IMDB	MAM, MDM	MDM, MAM, MDMDM	DM, AM, MDM

如表 3 所示，它展示了文献中广泛使用的预定义 meta-paths，以及 GTN 学习的具有高注意力分数的 meta-paths。可以看到 GTN 发现了不在预定义元路径集中的重要元路径。例如，在 DBLP 数据集中，GTN 将 CPCPA 列为最重要的元路径，而这并不包含在预定义的元路径集中。【作者的研究领域（用于预测的标签）与作者发表的地点有关是很有意义的。】由此我们可以认为，GTN 的可解释性通过元路径上的注意力得分为节点分类提供了有效的信息。



而上图展示了每个 GT 层的邻接矩阵的注意力分数, (a) 为 DBLP, (b) 为 IMDB。从中我们可以发现, 与 DBLP 相比, 单位阵  $I$  在 IMDB 中有更高的注意力分数。通过给单位阵  $I$  分配更高的注意力分数, GTN 试图坚持更短的元路径, 即使在更深的层。这表明了 GTN 能根据数据集自适应学习最有效的元路径的能力。

### 结论:

本文提出了用于学习异构图上节点表示的图变换网络 GTN。该方法将一个异构图转换为由任意边类型和任意长度的元路径定义的多个新图, 这些元路径的长度不超过 Graph Transformer 层的数量, 同时通过对学习到的元路径图进行卷积学习节点表示。实验表明学习到的图结构能导致更有效的节点表示, 在没有任何来自领域知识的预定义元路径的情况下, 在异构图上的所有三个基准节点分类任务上取得了最优性能。此外, 由于 Graph Transformer 层可以与现有的 GNNs 结合, 因此 GTN 框架能够为 GNNs 开辟新的途径, 根据数据和任务来操作卷积实现图结构的优化, 而不需要任何人工的努力。

### 对本文的感悟:

本文提出了一个简单却充满创新性且有效的改进, 它有效解决了人工设计“元路径”所带来的问题, 也再次表明了端到端学习的可行性和有效性。而究其原因, 我觉得这主要是在于作者们看清了“元路径”设计的本质数学形式, 进而提出了“元路径”自学习的设计方案——GTN。