

# 周报（2023.10.24-2023.11.7） 姓名：孙瑞阳

## 每日小结

	周一	周二	周三	周四	周五
早	重读 related work 文献	修改论文	重读 related work 文献	Graph 代码	Graph 代码
中	修改论文	写周报，阅读论文	重读 related work 文献	重读 related work 文献	修改论文
晚	重读 related work 文献	组会		Graph 代码	重读 related work 文献
	周一	周二	周三	周四	周五
早	修改论文	修改论文	重写 related work	重写 related work	Graph 代码
中	重写 related work	写周报，阅读论文	修改论文	补充的 Discussion 代码	补充的 Discussion 代码
晚	重写 related work	组会	组会	Graph 代码	补充的 Discussion 代码

注：简单表述当前时间段工作，如看文献1，整理数据等

## 科研详情

### 文献阅读

#### 文献 1

题目：Can Transformer and GNN Help Each Other?

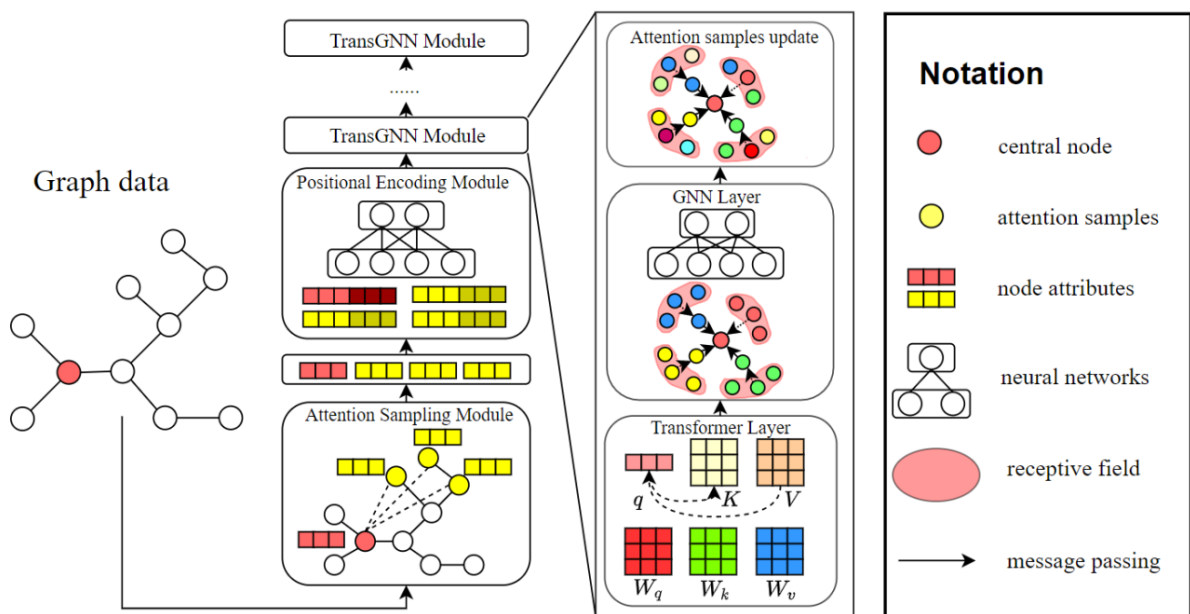
作者：Peiyan Zhang ,Senzhang Wang, Yuchen Yan, Sunghun Kim

出处：arxiv2023

方法：

提出了一种名为 **TransGNN** 的新模型，其中 **Transformer** 层和 **GNN** 层交替使用以相互改进。具体来说，为了扩大感受野并解开边的信息聚合，我们建议使用 **Transformer** 聚合更多相关节点的信息，以改善 **GNN** 的消息传递。此外，为了捕获图结构信息，我们利用位置编码并利用 **GNN** 层将结构融合为节点属性，从而改进了图数据中的 **Transformer**。

TransGNN 的框架如图：



该框架由三个重要组成部分组成: (1)注意力采样模块, (2)位置编码模块, (3) TransGNN 模块。首先通过考虑注意力采样模块中的语义相似度和图结构信息, 对每个中心节点进行最相关节点的采样。然后在位置编码模块中, 计算位置编码以帮助 Transformer 捕获图拓扑信息。

在这两个模块之后使用 TransGNN 模块, 该模块按顺序包含三个子模块: (i) Transformer 层, (ii) GNN 层, (iii)样本更新子模块。其中, Transformer 层用于扩展 GNN 层, 高效地聚合注意力样本信息, 而 GNN 层则帮助 Transformer 层感知图结构信息, 获取更多邻近节点的相关信息。样本更新子模块用于在新表示时有效地更新注意力样本。

## 节点分类

	PubMed	Amazon-Photo	Amazon-Computer	OGB-Arxiv
GCN	83.79±0.34%	90.46±0.17%	87.20±0.23%	71.74±0.29%
GraphSAGE	83.81±0.25%	90.63±0.31%	86.93±0.37%	71.21±0.27%
GAT	83.02±0.12%	90.69±0.08%	87.44±0.21%	71.49±0.12%
SGCN	82.79±0.63%	90.02±0.43%	87.31±0.17%	71.95±0.21%
GCNII	84.89±0.49%	91.91±0.27%	88.12±0.08%	72.74±0.16%
APPNP	85.02±0.35%	91.66±0.46%	87.01±0.21%	71.78±0.19%
GPRGNN	84.56±0.45%	92.91±0.26%	88.29±0.44%	71.76±0.23%
Graph-Bert	83.91±0.37%	91.88±0.40%	86.80±0.36%	71.32±0.34%
TransSAGE-RW	86.36±0.51%	93.63±0.49%	88.88±0.42%	72.12±0.41%
TransSAGE-MP	86.92±0.45%	93.89±0.36%	<b>90.20±0.35%</b>	72.33±0.22%
TransGCN-RW	86.23±0.26%	92.95±0.58%	89.34±0.33%	72.61±0.34%
TransGCN-MP	<b>87.53±0.37%</b>	<b>94.05±0.29%</b>	90.07±0.47%	<b>73.45±0.41%</b>
TransGAT-RW	86.58±0.35%	92.16±0.37%	89.92±0.61%	71.59±0.26%
TransGAT-MP	87.19±0.25%	93.36±0.25%	89.66±0.28%	71.87±0.23%

**Table 3: Performance of node classification task. The best result is bolded.**

启发:

1. 有代码, 在学习代码
2. 在节点分类的对比实验用到的对比模型, 也都可以学习一下。

## 文献2

题目: Do Transformers Really Perform Bad for Graph Representation?

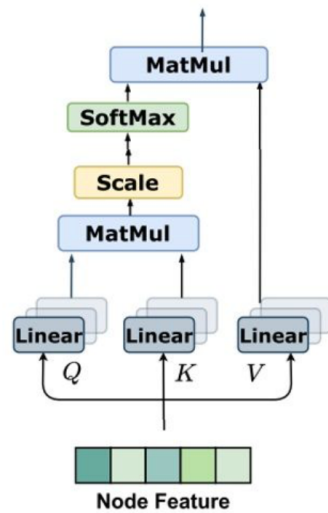
作者: Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, Tie-Yan Liu

出处: arxiv2023

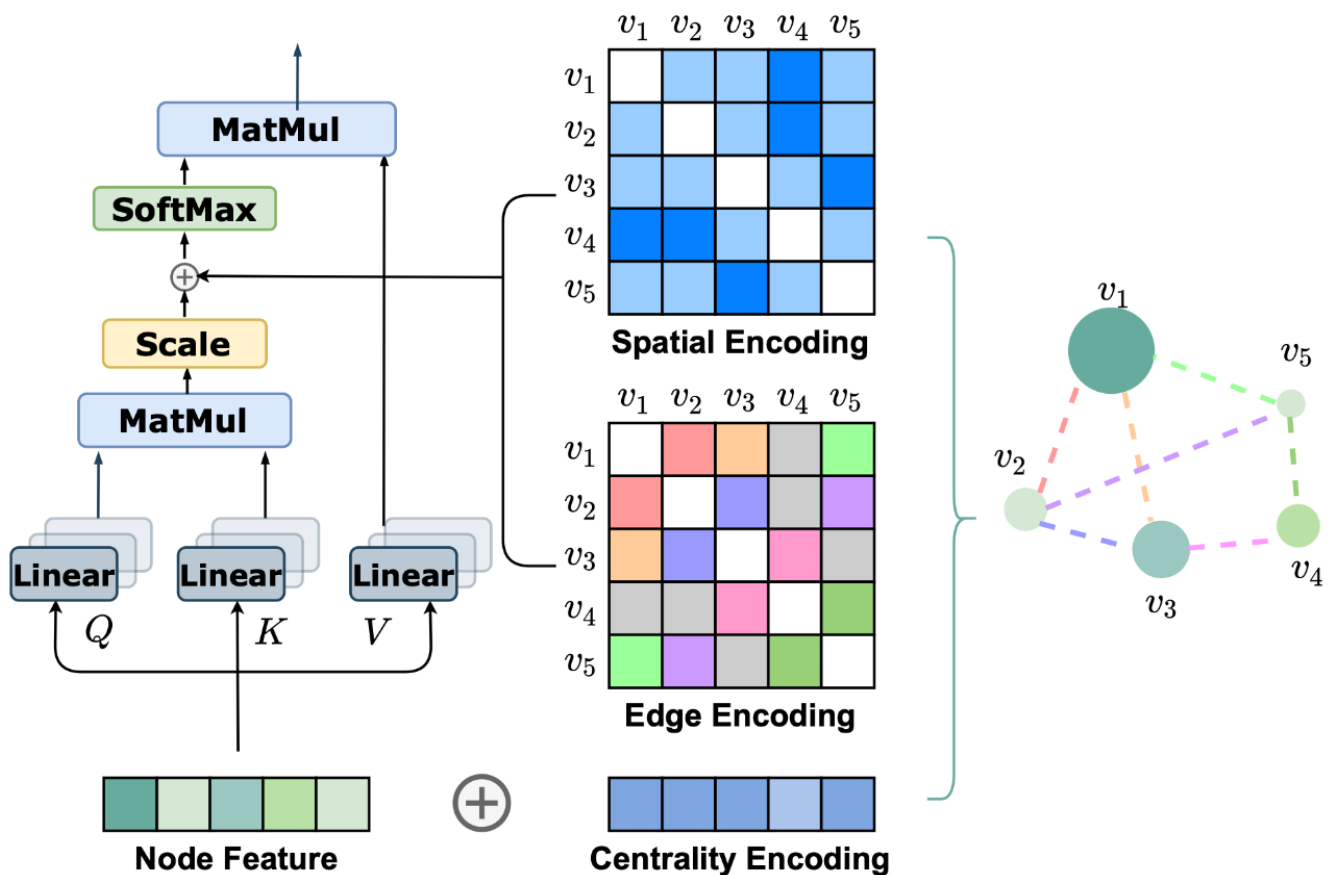
方法:

本文介绍了一种名为 Graphormer 的基于 Transformer 架构的模型, 专门用于图数据的表示学习。作者提出了结构信息编码方法, 包括中心性编码和空间编码, 以有效地捕捉图中的节点重要性和节点之间的结构关系。

如果直接对节点做 QKV (如图), 只能得到节点特征的语义特征相似, 丢失了图的结构信息:



作者提出的 Graphormer（如图），能够保留图的结构信息，并抓取节点特征的语义相似：



为了有效地将图的结构信息融入到 Transformer 模型中，作者提出了几种结构编码方法。其中主要的编码包括：

- **中心性编码**：捕捉节点的重要性。
- **空间编码**：捕捉节点之间的结构关系。

启发：

1. 关于中心性编码和空间编码的部分偏向于公式解析，没看太懂。

但大致思路可以借鉴：

图的节点级Transformer = 节点的QKV信息 + 图结构信息

在Transformer的结构中有效利用图的结构信息是重点

## 工作进展

1: 重看了相关领域的论文，重写了related work，写完了

还差实验分析重写，写完给您

2: 增加 3 个 discussion，充实一些关于模型可解释性和时空模式的分析，代码差不多写完了，结果还在筛选

## 三个discussion补充

- 增加一个自注意力融合前后的社交和遥感特征的区别对比，证明自注意力融合互补了特征（混淆矩阵？或者特征可视化）
- 增加不同功能区的注意力更关注的时序区间：输出注意力平均值就行，看乐阳老师组的工作
- 补充关于社交网络卷积transformer 的卷积核k的选取讨论：实验验证k为24时精度最高（或者注意力得分之类的最高），符合社交数据在24小时时域尺度上的时序特征最显著。

3:建立 graph:

重新使用 python 的 numpy 和 pandas 库清洗用户访问数据，

## 下周计划

1. 多思考，增加 discussion，增加一些社交数据时空模式的分析
2. 找长时序 graph，直接对节点做分类的相关文献
3. 数据清洗代码和建 graph 代码