

Combine GNNs with LLMs

MSC技术分享-2024-06-01

温富康

GNNs

- 图神经网络是一种特殊的神经网络
- 和传统的MLP相比，图神经网络依赖一个特定的图结构进行消息的传递和聚合
- 图神经网络遵循的一般范式可以描述成如下的形式：

$$h_i^{(l)} = \mathbf{U}\left(h_i^{(l-1)}, \mathbf{M}(\{h_i^{(l-1)}, h_j^{(l-1)} | v_j \in \mathcal{N}_i\})\right)$$

- 图神经网络的应用：节点级，边级，图级，...

LLMs

- 大语言模型是参数特别大的语言模型
- LLMs在处理文本的领域已经展示出了非凡的威力
- LLMs具有很好的zero-shot能力
- 常见的大语言模型：
 - 闭源：ChatGPT4o, ChatGPT4, ...
 - 开源：LLaMa2, ...

LLMs训练/学习/使用的范式

- Pre-Training
- Fine-Tuning
- Prompting

LLMs + GNNs的工作分类

- 使用LLMs进行增强
- 使用LLMs进行猜测
- 让LLMs和GNNs对齐
- 其他工作

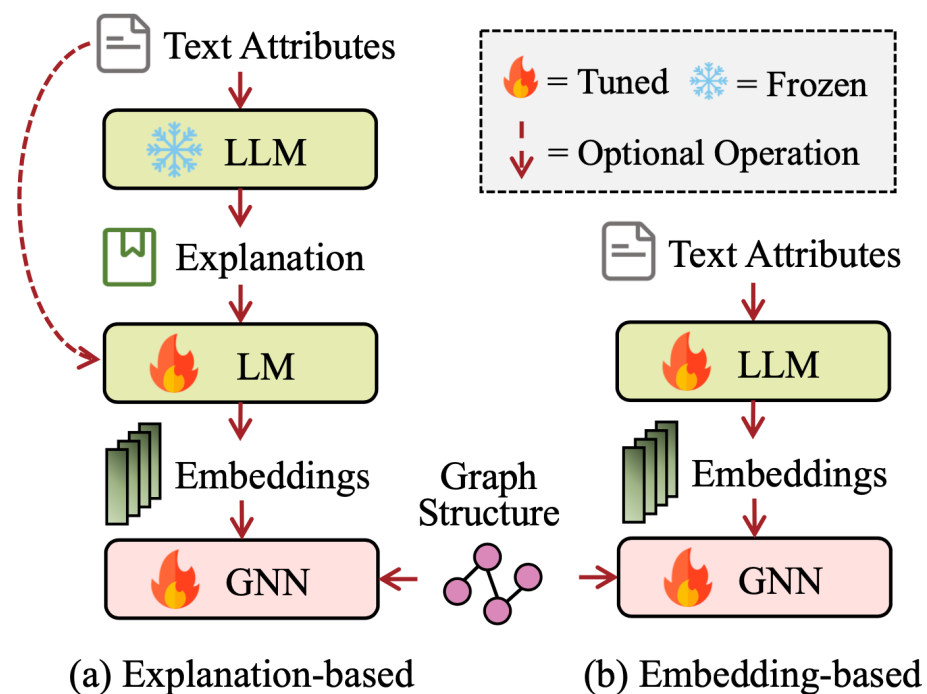
使用LLMs进行增强

- 一些数据集中包含了文本特征
 - Cora
- 传统的技术对于文本特征的嵌入过于初等
- LLMs可以用于提升结点嵌入的质量
- 根据LLMs是否生成新的文本又可以分成以下两类：
 - 基于解释的LLMs增强器
 - 基于嵌入的LLMs增强器

基于解释的LLMs增强方法

- 不仅会处理文本特征，还会产生新的文本
- 这些新的文本被称为解释
- LLMs可以捕捉到更高阶的含义和信息

基于解释的LLMs增强方法



基于解释的LLMs增强方法

- 基于解释的方法可以形式化的描述成：

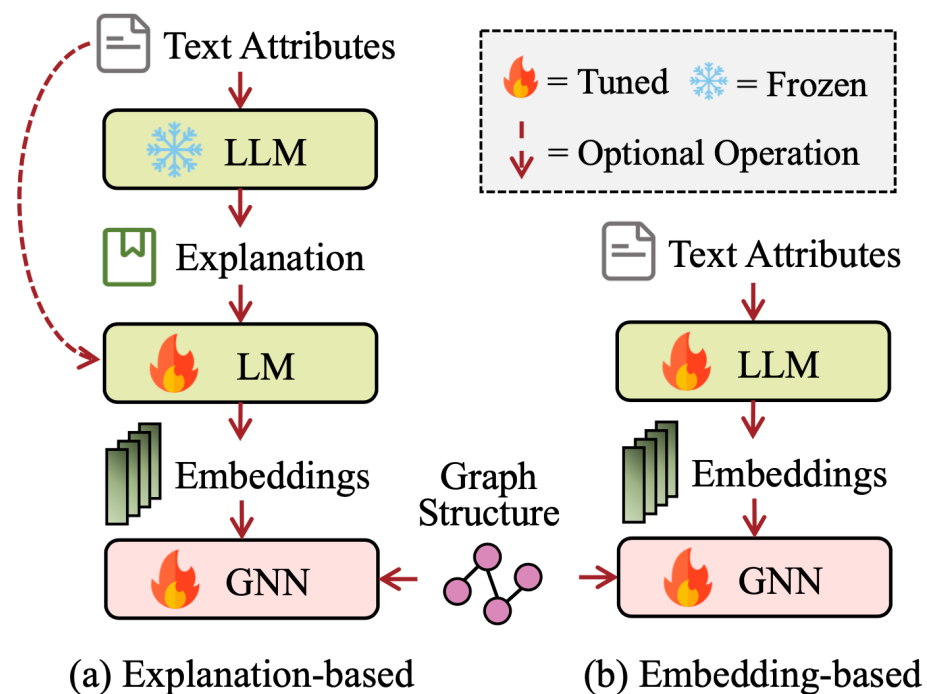
Enhancement: $e_i = f_{\text{LLM}}(t_i, p)$, $\mathbf{x}_i = f_{\text{LM}}(e_i, t_i)$,

Graph Learning: $\mathbf{H} = f_{\text{GNN}}(\mathbf{X}, \mathbf{A})$,

基于嵌入的LLMs增强方法

- 不产生新的文本特征
- 直接对节点的文本特征进行嵌入

基于嵌入的LLMs增强方法



基于嵌入的LLMs增强方法

- 基于嵌入的方法可以形式化的描述成：

$$\text{Enhancement: } \mathbf{x}_i = f_{\text{LLM}}(t_i),$$

$$\text{Graph Learning: } \mathbf{H} = f_{\text{GNN}}(\mathbf{X}, \mathbf{A}).$$

- 基于嵌入的LLMs增强方法必须使用开源的或者嵌入可见的LLMs
 - 我们可以使用开源的Llama2模型
 - 我们不可以使用GPT4因为得不到文本特征被嵌入后的向量

使用LLMs进行增强

- 优点

- 高效且精准的捕捉数据集的文本信息和结构信息
- 强大的灵活性：LLMs和GNNs是独立的
- 基于解释的增强方式能够利用闭源的LLMs

- 缺陷

- 处理较大规模的图的时候开销很巨大(GPT4-API价格昂贵)

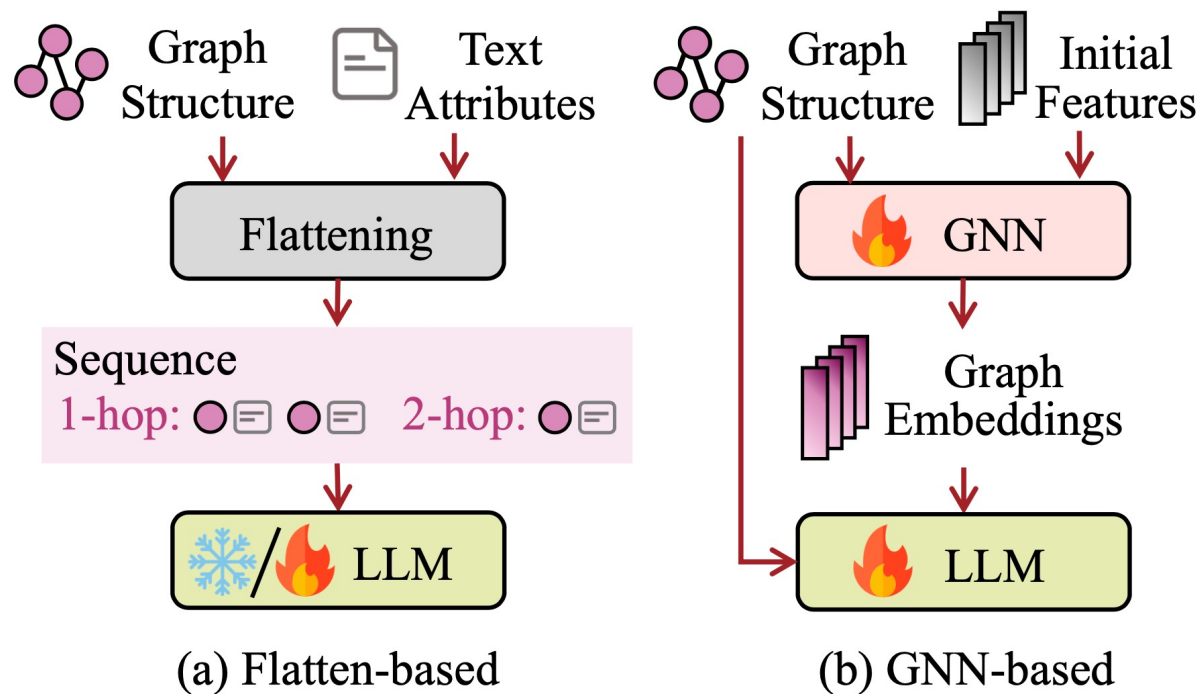
使用LLMs进行预测

- LLMs直接成为预测器
- 根据是否使用GNNs为LLMs提取结构信息又可以分成两类：
 - 基于展开的LLMs预测器
 - 基于GNNs的LLMs预测器

基于展开的LLMs预测器

- LLMs只能处理文字信息
- LLMs无法直接理解图结构
- 我们将图结构转换成LLMs可以理解的形式，这一过程称为展开

基于展开的LLMs预测器



基于展开的LLMs预测器

- 基于展开的LLMs预测器可以形式化的描述成：

Graph Flattening: $G_{seq} = \text{Flat}(\mathcal{V}, \mathcal{E}, \mathcal{T}, \mathcal{J}),$

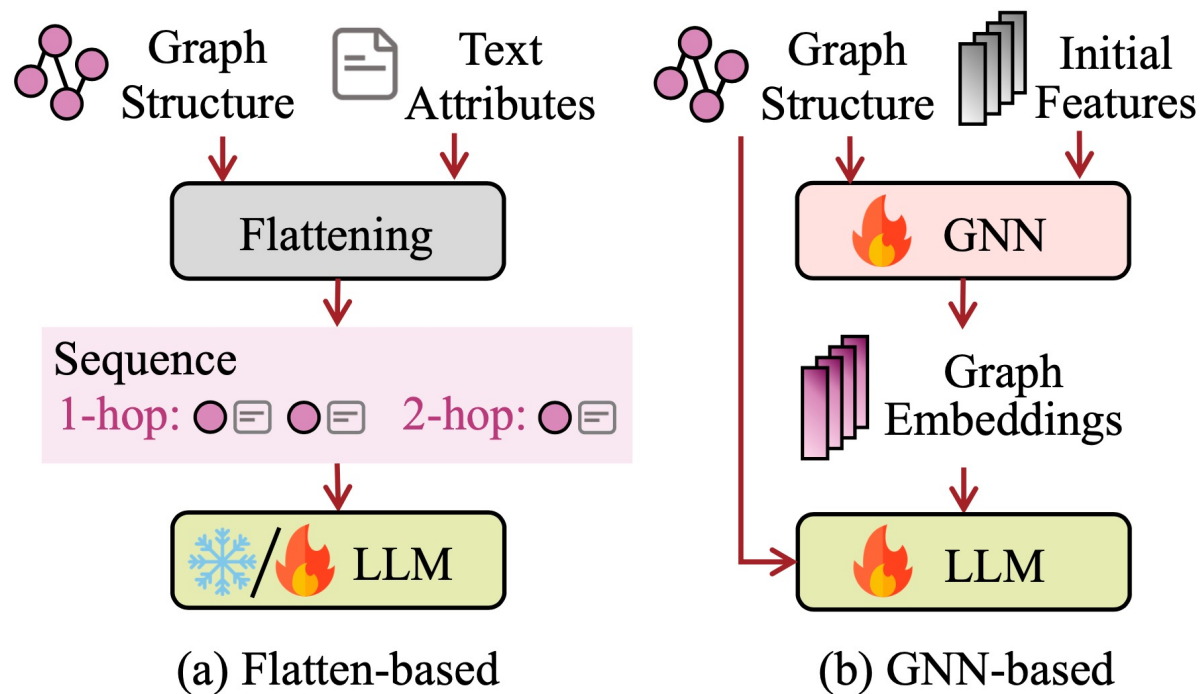
Prediction: $\tilde{Y} = \text{Parse}(f_{\text{LLM}}(G_{seq}, p)),$

- 解析过程的一些技巧：
 - 使用正则表达式
 - 设置decoder的温度为0，降低结果的不稳定性
 - 将解析过程转化成一个Q-A过程
 - ...

基于GNNs的LLMs预测器

- GNNs真的能够很好的理解图的结构
- 使用GNNs提炼图的结构信息给LLMs进行后续的工作

基于GNNs的LLMs预测器



基于GNNs的LLMs预测器

- 基于GNNs的LLMs预测器可以形式化的描述成：

$$\text{Graph Learning: } \mathbf{H} = f_{\text{GNN}}(\mathbf{X}, \mathbf{A}),$$

$$\text{Prediction: } \tilde{Y} = \text{Parse}(f_{\text{LLM}}(\mathbf{H}, p)),$$

- 一些工作：
 - 将图结构映射称为文本信息
 - 使用图Transformer将图的结构化信息传递给LLMs
 - ...

使用LLMs进行预测

- 优点
 - 能够很好的利用LLMs的zero-shot的能力
- 缺陷
 - LLMs对输入长度有限制导致难以处理大规模的图
 - 引入LLMs并不能解决一些GNNs固有的问题(例如异配图)

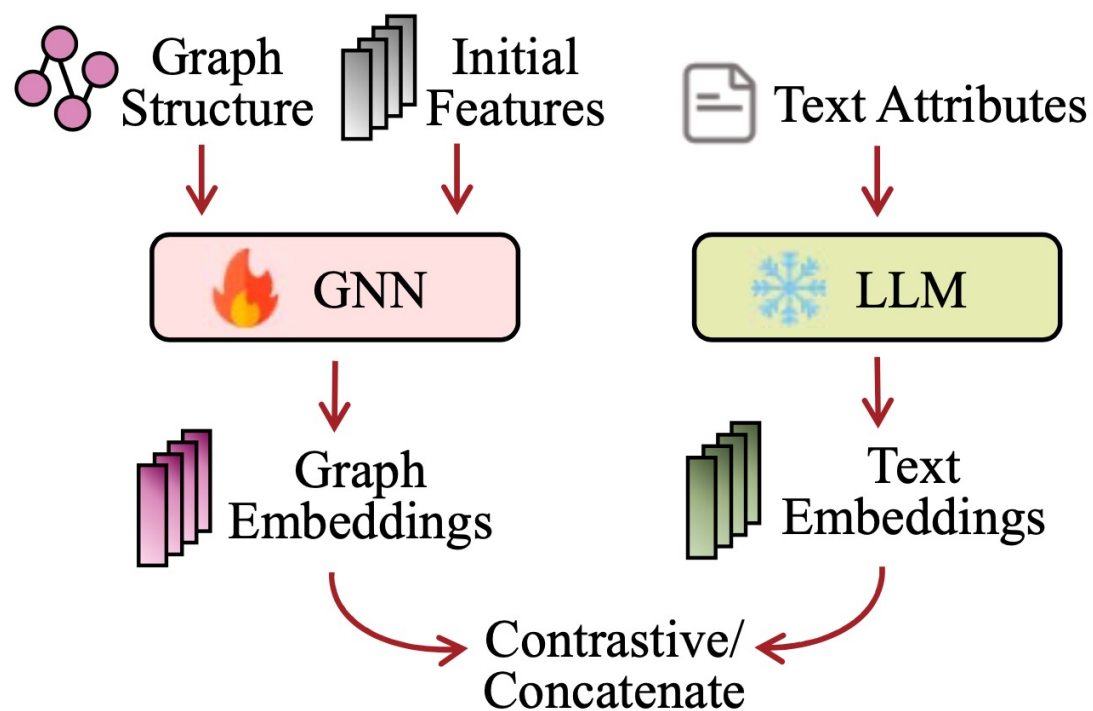
让LLMs和GNNs对齐

- 我们将LLMs和GNNs视为两种不同的模态
- 让两种模态相互配合工作
- 根据其中一种模态是否比另一种模型更强工作可以分成两类：
 - 对称的GNNs-LLMs对齐
 - 非对称的GNNs-LLMs对齐

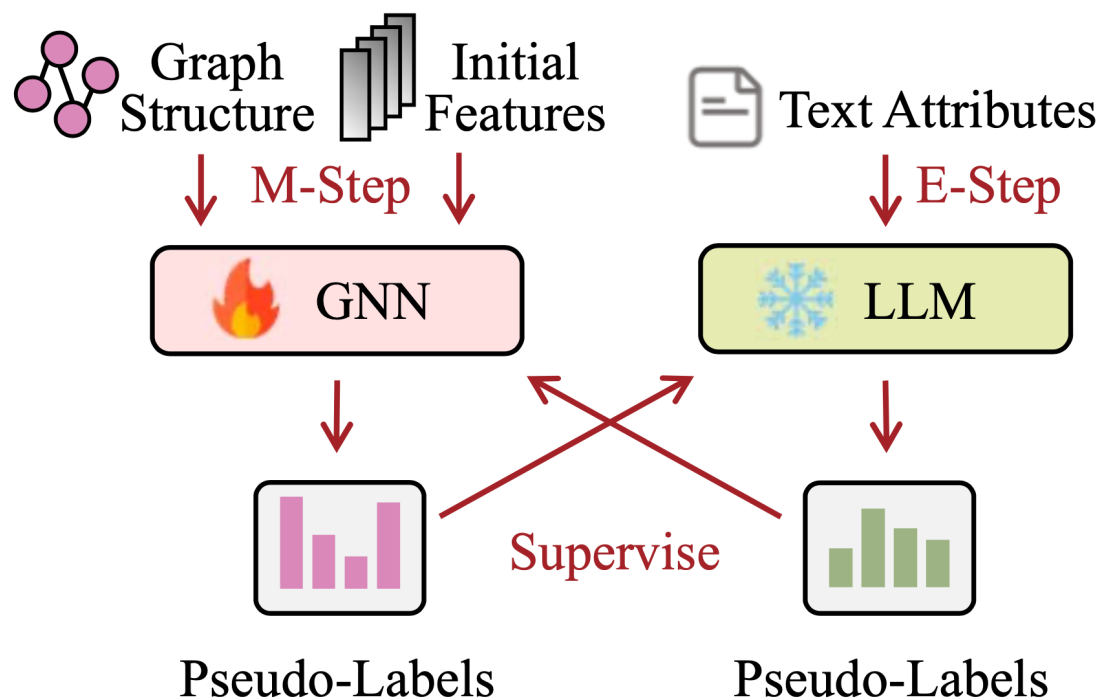
对称的GNNs-LLMs对齐

- 我们认为LLMs和GNNs地位等同
- 使用两个模态进行对比学习
- 可以使用单次或者多次迭代

对称的GNNs-LLMs对齐



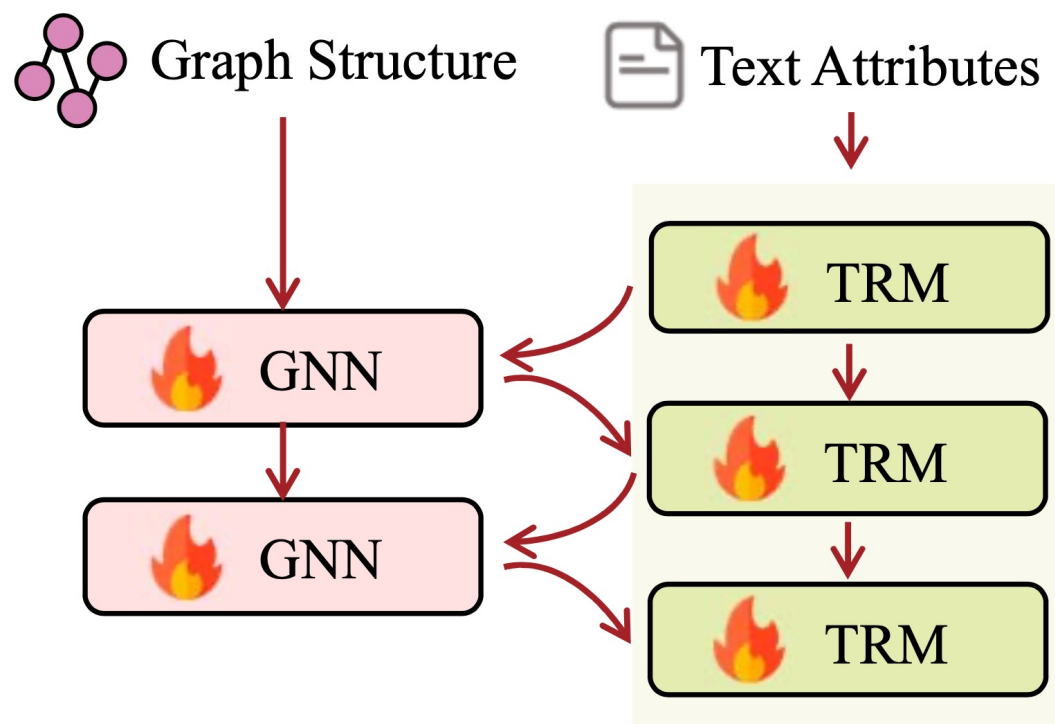
对称的GNNs-LLMs对齐



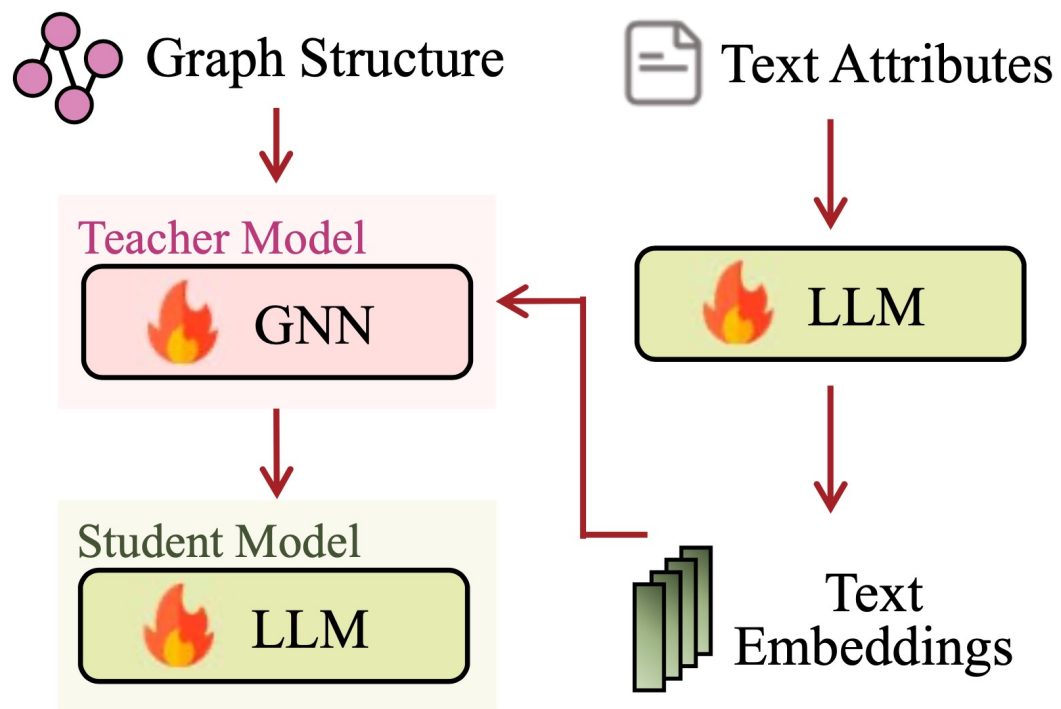
非对称的GNNs-LLMs对齐

- LLMs和GNNs中有一个模型更加重要
- 可以使用以下两种方式让GNNs去增强LLMs:
 - Graph-Nested Transformer
 - Graph-Aware Distillation

Graph-Nested Transformer



Graph-Aware Distillation



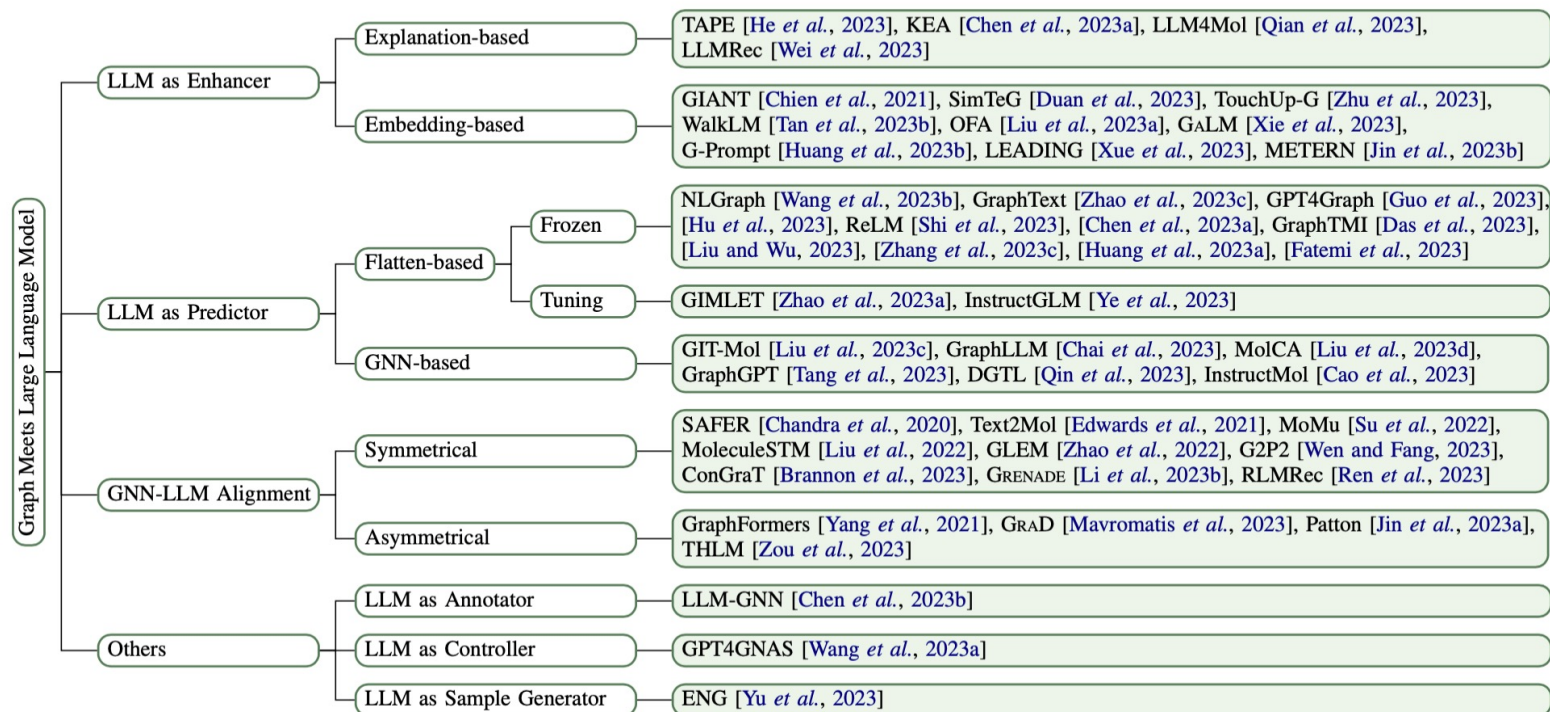
让LLMs和GNNs对齐

- 优点
 - 能够处理两个模态的任务
- 缺陷
 - 缺少足够的Text-Attributed Graph的数据集

一些未来研究方向

- 处理non-TAG
 - 并非所有图数据集都是Text-Attributed-Graph
- 处理数据泄漏
 - LLMs在训练过程中已经见过数据集本身
- 增强可迁移性
 - 领域之间的知识/图结构能否进行迁移
- 提高效率
 - LLMs的训练和部署消耗很多的硬件资源
- 分析表达能力
 - LLMs到底能够理解多少图结构

一些更具体的文章



希望MSC的各位朋友们儿童节快乐！