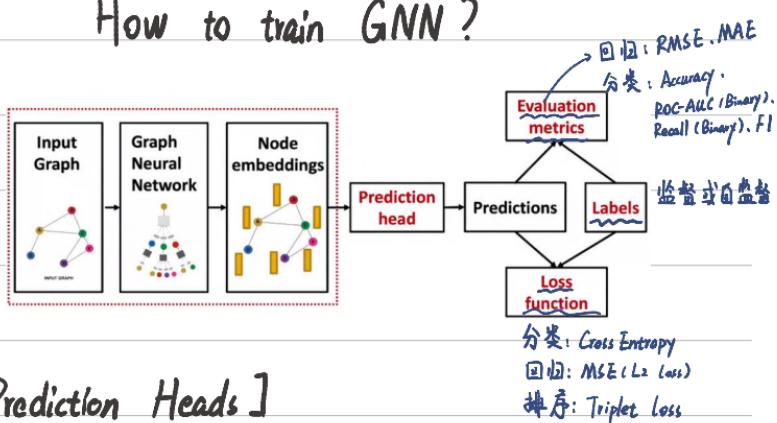


日期: /

How to train GNN?



[GNN Prediction Heads]

对于不同级别的GNN,有不同的预测头。

1) 对于节点级的任务,直接用节点嵌入产生预测。

$$\hat{y}_v = \text{Head}_{\text{node}}(h_v^{(L)})$$

2) 对于边级的任务,使用节点嵌入对产生预测。

$$\hat{y}_{uv} = \text{Head}_{\text{edge}}(h_u^{(L)}, h_v^{(L)})$$

3) 对于图级别的任务,使用图中所有嵌入进行预测。

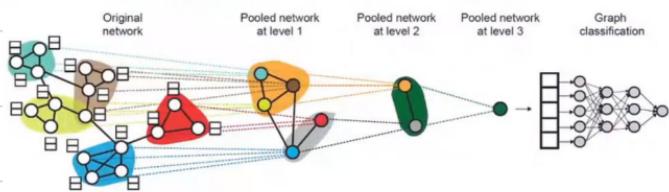
$$\hat{y}_G = \text{Head}_{\text{graph}}(\{h_v^{(L)} \in \mathbb{R}^d, \forall v \in G\})$$

可选 mean pooling, max pooling, sum pooling ...

一个更有效的策略是使用层次化的全局池化,即分批对图中的所有节点嵌入进行聚合。

日期： /

[Hierarchical Pooling (DiffPool)]



每一层次的池化中包含 2 个独立的 GNNs：

- ① 用于计算节点嵌入
- ② 用于对节点进行聚类

[Metrics for Binary Classification]

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

$$\text{Precision (P)} = \frac{TP}{TP + FP}$$

$$\text{Recall (R)} = \frac{TP}{TP + FN}$$

$$\text{F1-Score} = \frac{2PR}{P+R}$$

ROC-AUC : ROC 曲线以假正类率 ($FP/(FP+TN)$)

为横轴, 以召回率为纵轴, ROC-AUC 为 ROC 下

面积, 其值越接近 1, 代表模型分类效果越好。

ROC-AUC=1 时代表所有正例的预测概率均大于负例, 无论阈值为何
模型总能进行有效区分。