Masked Label Prediction: Unified Message Passing Model for Semi-Supervised Classification

https://arxiv.org/pdf/2009.03509.pdf

研究背景：

有两种在图上传递信息的模型：Graph Neural Networks（图神经网络）和Label Propagation Algorithm （标签传递算法）。前者主要传递的是节点的特征，后者传递的是节点的标签。其中GCN/GAT/GraphSAGE/SGC 都输入前者，而PageRank，Personalized PageRank等都属于后者。也有一些模型整合了前后两种模型，在图神经网络中加入了标签传递层，比如GCN-LPA，APPNP等。但作者认为：之前的模型知识拼接起来了两种方法，而没有把两种方法统一到一个模型中。在训练和预测都使用节点特征和真实标签。所以作者提出了UniMP模型，其优点如下：

- 在GNN模型训练中和预测时都使用了真实标签和节点特征
- 受启发于Bert中的MLM，提出Masked Label Prediction，即遮盖住一部分训练节点的真实标签，然后通过特征去预测这些标签。

模型结构如下：

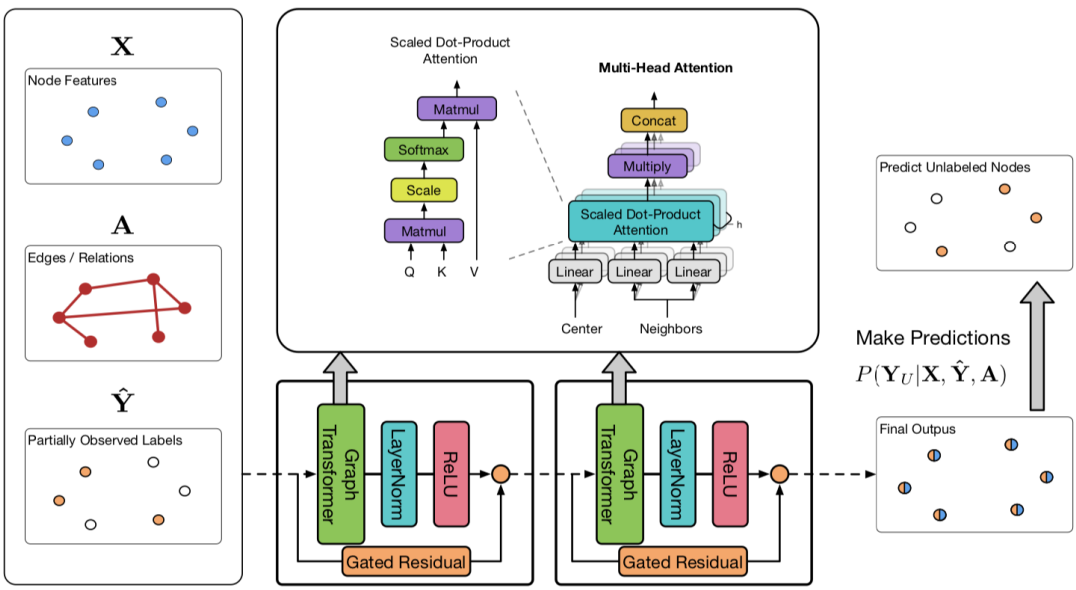

Figure 1: The architecture of UniMP.

输入为节点的特征向量X，邻接矩阵A和真实标签向量Y。

- Graph Transformer

  借鉴于Attention is all you need一文中的attention结构，作者提出了使用muti-head attention的graph transformer结构。其中

$$q_{c,i}^{(l)} = W_{c,i}^{(l)} h_i^{(l)} + b_{c,q}^{(l)}$$

$$k_{c,j}^{(l)} = W_{c,j}^{(l)} h_i^{(l)} + b_{c,k}^{(l)}$$

$$e_{c,ij} = W_{c,e} e_{ij} + b_{c,e}$$

$$\alpha_{c,ij}^{(l)} = \frac{\langle q_{c,i}^{(l)}, k_{c,i}^{(l)} + e_{c,ij} \rangle}{\sum_{u \in \mathcal{N}(i)} \langle q_{c,u}^{(l)}, k_{c,u}^{(l)} + e_{c,iu} \rangle}$$

第一行为源点过一层FC，第二行为汇点过一层FC，第三行为源点汇点之间的边权值过一层FC，最后对这些值做dot product attention。最后在每一层中，汇点的特征向量经过FC之后加权，此处权值为汇点到源点的attention（第四行）。如果是多头注意力，每一个注意力的输出channel数量是总的输出channel/多头注意力的头数。这样多头注意力没有降低计算效率。

我们对比UniMP中的注意力和GAT中的注意力，可以看出GAT使用的是拼接（concat），而UniMP使用的是向量点积。下图为GAT中的注意力。

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

- Gated Residual Connection & output attention

  作者使用了带门控的残差连接来防止过平滑；同时如果想要在最后一层使用attention，不是拼接每一个『头』，而是将每个头取平均。这样使得输出channel数量小。

- LPA

  作者首先通过embedding，将标签向量映射到和特征向量一样的维度，如果没有标签则是个零向量。而后作者在使用标签时，直接将节点标签向量Y与特征向量X相加，注意不是拼接，是相加。这样保持输入维度不变。这样每一层的hidden state H更新公式如下：

$$H^{(0)} = X + \hat{Y} W_{emb}$$

$$H^{(l+1)} = \sigma(((1-\beta)A + \beta I) H^{(l)} W^{(l)})$$

  其中beta为残差连接的门控参数，是学习出来的。

- Masked Label Prediction

为了防止真实标签泄露给训练数据，受启发于NLP中BERT模型的MLM训练任务，作者提出了遮盖住一部分真实标签，把他们设置为0，这样用于训练。而在测试时，所有真实标签都不被遮盖。

实验结果：

首先作者在三大ogbn-（protein，arxiv，products）的数据集中都打败了sota模型。

Model	Test Accuracy	Validation Accuracy	Params
GCN-Cluster [Chiang *et al.*, 2019]	0.7897 ± 0.0036	0.9212 ± 0.0009	206,895
GAT-Cluster	0.7923 ± 0.0078	0.8985 ± 0.0022	1,540,848
GAT-NeighborSampling	0.7945 ± 0.0059	-	1,751,574
GraphSAINT [Zeng *et al.*, 2019]	0.8027 ± 0.0026	-	331,661
DeeperGCN [Li *et al.*, 2020]	0.8090 ± 0.0020	0.9238 ± 0.0009	253,743
UniMP	**0.8256 ± 0.0031**	**0.9308 ± 0.0017**	1,475,605

Table 4: Results for ogbn-products

Model	Test ROC-AUC	Validation ROC-AUC	Params
GaAN [Zhang *et al.*, 2018]	0.7803 ± 0.0073	-	-
GeniePath-BS [Liu *et al.*, 2020b]	0.7825 ± 0.0035	-	316,754
MWE-DGCN	0.8436 ± 0.0065	0.8973± 0.0057	538,544
DeepGCN [Li *et al.*, 2019]	0.8496 ± 0.0028	0.8921 ± 0.0011	2,374,456
DeeperGCN [Li *et al.*, 2020]	0.8580 ± 0.0017	0.9106 ± 0.0016	2,374,568
UniMP	**0.8642 ±0.0008**	**0.9175 ± 0.0007**	1,909,104

Table 5: Results for ogbn-proteins

Model	Test Accuracy	Validation Accuracy	Param
DeeperGCN [Li *et al.*, 2020]	0.7192 ± 0.0016	0.7262 ± 0.0014	1,471,506
GaAN [Zhang *et al.*, 2018]	0.7197 ± 0.0024	-	1,471,506
DAGNN [Liu *et al.*, 2020a]	0.7209 ± 0.0025	-	1,751,574
JKNet [Xu *et al.*, 2018b]	0.7219 ± 0.0021	0.7335 ± 0.0007	331,661
GCNII [Chen *et al.*, 2020]	0.7274 ± 0.0016	-	2,148,648
UniMP	**0.7311 ± 0.0021**	**0.7450 ± 0.0005**	473,489

Table 6: Results for ogbn-arxiv

其次作者验证标签传播（LPA）确实有助于提高准确率。

Inputs	Model	Datasets		
		ogbn-products Test ACC	ogbn-proteins Test ROC-AUC	ogbn-arxiv Test ACC
$\mathbf{X}$	Multilayer Perceptron	0.6106 ± 0.0008	0.7204 ± 0.0048	0.5765 ± 0.0012
$\mathbf{X}, \mathbf{A}$	GCN	0.7851 ± 0.0011	0.8265 ± 0.0008	0.7218 ± 0.0014
	GAT	0.8002 ± 0.0063	0.8376 ± 0.0007	0.7246 ± 0.0013
	Graph Transformer	0.8137 ± 0.0047	0.8347 ± 0.0014	0.7292 ± 0.0010
$\mathbf{A}, \hat{\mathbf{Y}}$	GCN	0.7832 ± 0.0013	0.8083 ± 0.0021	0.7018 ± 0.0009
	GAT	0.7751 ± 0.0054	0.8247 ± 0.0033	0.7055 ± 0.0012
	Graph Transformer	0.7987 ± 0.0104	0.8160 ± 0.0007	0.7090 ± 0.0007
$\mathbf{X}, \mathbf{A}, \hat{\mathbf{Y}}$	GCN	0.7987 ± 0.0104	0.8247 ± 0.0032	0.7264 ± 0.0003
	GAT	0.8193 ± 0.0017	0.8556 ± 0.0009	0.7278 ± 0.0009
	Graph Transformer	**0.8256 ± 0.0031**	0.8560 ± 0.0003	**0.7311 ± 0.0021**
	∟ w/ Edge Feature	*	**0.8642 ± 0.0008**	*

最后作者用腐化实验（ablation studies）验证UniMP中的一些tricks确实有效果。

Model	ogbn-prdouct	ogbn-arxiv
GAT (sum attention)	0.8002	0.7246
∟ w/ residual	0.8033	0.7265
∟ w/ gated residual	0.8050	0.7272
Transformer (dot-product)	0.8091	0.7259
∟ w/ residual	0.8125	0.7271
∟ w/ gated residual	0.8137	0.7292
∟ w/ train label (UniMP)	0.8256	0.7311
∟ w/ validation labels	**0.8312**	**0.7377**