

SEMI-SUPERVISED CLASSIFICATION WITH GRAPH CONVOLUTIONAL NETWORKS 笔记

Introduction

本篇文章的主要工作是将卷积扩展到图结构的数据中，能够得到比较好的数据表示，并且在半监督任务中也取得了不错的效果。该网络是传统卷积算法在图结构数据上的一个变体，可以直接用于处理图结构数据。从本质上讲，GCN 是谱图卷积（spectral graph convolution）的局部一阶近似（localized first-order approximation）。GCN 的另一个特点在于其模型规模会随图中边的数量的增长而线性增长。总的来说，GCN 可以用于对局部图结构与节点特征进行编码。使用神经网络模型 $f(X, A)$ 对所有带标签节点进行基于监督损失的训练。其中， X 为输入数据， A 为图的邻接矩阵。在图的邻接矩阵上调整 $f(\cdot)$ 将允许模型从监督损失 L_0 中分配梯度信息，并使其能够学习所有节点（带标签或不带标签）的表示。

图卷积神经网络

本文得到了图卷积神经网络的（单层）最终形式：

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}\right). \quad (2)$$

Here, $\tilde{A} = A + I_N$ is the adjacency matrix of the undirected graph \mathcal{G} with added self-connections. I_N is the identity matrix, $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ and $W^{(l)}$ is a layer-specific trainable weight matrix. $\sigma(\cdot)$ denotes an activation function, such as the $\text{ReLU}(\cdot) = \max(0, \cdot)$. $H^{(l)} \in \mathbb{R}^{N \times D}$ is the matrix of activations in the l^{th} layer; $H^{(0)} = X$. In the following, we show that the form of this propagation rule can be motivated¹ via a first-order approximation of localized spectral filters on graphs (Hammond et al., 2011; Defferrard et al., 2016).

Model

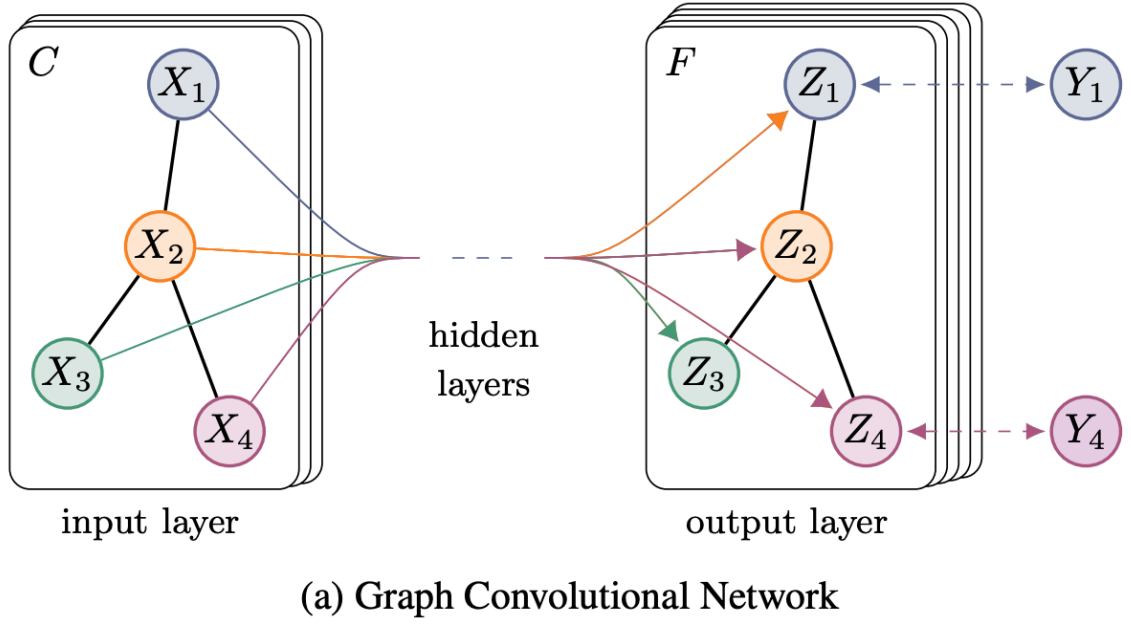
对于一个大图（例如“文献引用网络”），我们有时需要对其上的节点进行分类。然而，在该图上，仅有少量的节点是有标注的。此时，我们需要依靠这些已标注的节点来对那些没有标注过的节点进行分类，此即半监督节点分类问题。在这类问题中，由于大部分节点都没有已标注的标签，因此往往需要使用某种形式的图正则项对标签信息进行平滑（例如在损失函数中引入图拉普拉斯正则）。

$$\mathcal{L} = \mathcal{L}_0 + \lambda \mathcal{L}_{\text{reg}}, \quad \text{with} \quad \mathcal{L}_{\text{reg}} = \sum_{i,j} A_{ij} \|f(X_i) - f(X_j)\|^2 = f(X)^\top \Delta f(X). \quad (1)$$

其中， \mathcal{L}_0 表示有监督的损失， $f(\cdot)$ 可以是一个类似于神经网络的可微函数。 λ 表示一个权值因子， X 则是相应的节点向量表示。 $\Delta = D - A$ 表示未归一化的图拉普拉斯矩阵。这种处理方式的一个基本假设是：相连的节点可能有相同的标签。然而，这种假设却往往会限制模型的表示能力，因为图中的边不仅仅可以用于编码节点相似度，而且还包含有额外的信息。

GCN的使用可以有效地避开这一问题。GCN通过一个简单的映射函数 $f(\mathbf{X}, \mathbf{A})$ ，可以将节点的局部信息汇聚到该节点中，然后仅使用那些有标注的节点计算 \mathcal{L}_0 即可，从而无需使用图拉普拉斯正则。

具体来说，本文使用了一个两层的GCN进行节点分类。模型结构图如下图所示：



其具体流程为：

- 首先获取节点的特征表示 \mathbf{X} 并计算邻接矩阵 $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}}$ 。
- 将其输入到一个两层的GCN网络中，得到每个标签的预测结果：

$$\mathbf{Z} = f(\mathbf{X}, \mathbf{A}) = \text{softmax}\left(\hat{\mathbf{A}} \text{ReLU}\left(\hat{\mathbf{A}} \mathbf{X} \mathbf{W}^{(0)}\right) \mathbf{W}^{(1)}\right). \quad (9)$$

其中， $\mathbf{W}^{(0)} \in \mathbb{R}^{C \times H}$ 为第一层的权值矩阵，用于将节点的特征表示映射为相应的隐层状态。 $\mathbf{W}^{(1)} \in \mathbb{R}^{H \times F}$ 为第二层的权值矩阵，用于将节点的隐层表示映射为相应的输出（ \mathbf{F} 对应节点标签的数量）。最后将每个节点的表示通过一个softmax函数，即可得到每个标签的预测结果。

对于半监督分类问题，使用所有有标签节点上的期望交叉熵作为损失函数：

$$\mathcal{L} = - \sum_{l \in \mathcal{Y}_L} \sum_{f=1}^F Y_{lf} \ln Z_{lf}, \quad (10)$$

其中，

$$\mathcal{Y}_L$$

表示有标签的节点集。