This is a note for the paper: Michaël Defferrard, Xavier Bresson, Pierre Vandergheynst, Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering, Neural Information Processing Systems (NIPS), 2016.

paper and code have been published by the author

问题背景

问题的抽象描述： 一个数据被表示为一张图，一个属性为一个节点，属性用一个向量$x \in R^d$表示，属性之间有某种权重（比如相似度）。希望对数据进行分类，进行监督学习。

问题的一个实例：图像是特殊的图数据，像素间的距离为$dist = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$,$(x_i, y_i)$为像素的空间坐标。像素间权重可以取$W_{ij} = exp(-\frac{dist^2}{2\sigma^2})$。对每个像素，只取最大的k个权重，其余置为0。

另一个实例：文本分类。采用词袋模型。这时所有的词构成一张图，每个词被一个embedding vector v_i表示。词之间的权重$W_{ij} = exp(-\frac{||v_i - v_j||_2^2}{2\sigma^2})$同样。对每个单词，只取最大的k个权重，其余置为0。

图上的卷积

一个图上的标准卷积是$g * x = U\Theta U^T x$ ，其中x为图上的一个信号，Θ为对角阵

计算量为$O(N^2)$ N为图中的点数

设拉普拉斯矩阵$L = U\Gamma U^T$

为了简化计算，注意到Θ为拉普拉斯矩阵特征值时$g * x = Lx$

设$\Theta \approx \sum_{i=1}^{K} \theta_i \Gamma^K$ (感受野大小为K)

则卷积$g * x = U\Theta U^T x \approx U \sum_{i=1}^{K} \theta_i \Gamma^K U^T x = \sum_{i=1}^{K} \theta_i L^K x$

计算$L^K x$时采用递推计算，并且一次递推是稀疏矩阵乘法，总复杂度为$O(KE)$,E为边数。由于是k近邻图，复杂度为$O(KkN)$ N为图中的点数

实际计算时利用了切比雪夫多项式，不过主要的思想如上。

由于输入是d个图上的信号，因此一个卷积滤波器为对d个信号做卷积再相加，卷积系数为可训练的参数。

如果有d1个卷积滤波器，则输出为图上的d1维信号。

图上的池化操作

本文中，池化操作其实是一种自底向上的层次聚类，每次将两个点聚在一起（绑定为一个节点），从而生成一个节点数大约减半的子图。选择绑定节点的方法是$max \quad W_{ij}(\frac{1}{d_i} + \frac{1}{d_j})$,每次固定i，选择使上述normalized cut最大的点j，然后绑定它们，作为一个新节点，新节点的权重为之前两个节点权重的和。（开源代码中优先对度小的点进行配对）

由此可以逐级定义子图，即对绑定的节点进行pooling操作。

我重点阅读了coarsening，即pooling的开源代码

卷积神经网络最重要的三个功能：局部感受野、参数共享、池化操作，在这篇图卷积网络中也就都实现了。

KIPF的GCN中每个输入的数据是一个节点，图规模随数据增加而增大，训练方法是半监督学习。而DEFFER的GCN中，每个输入的数据被建模为一个图，数据类型给定则图规模就定了，图规模不会随数据量增大而增大。训练方法是监督学习，很像传统的CNN。