深度学习导论 DS2001.01.2024SP——实验三

实验要求

使用 pytorch 或者 tensorflow 的相关神经网络库，编写图卷积神经网络模型 (GCN)，并在相应的图结构数据集上完成节点分类和链路预测任务，最后分析自环、层数、DropEdge 、PairNorm 、激活函数等因素对模型的分类和预测性能的影响 。

实验步骤

1. **网络框架**：要求选择 pytorch 或 tensorflow 其中之一，依据官方网站的指引安装包。（如果前面实验已经安装过，则这个可以跳过）
2. **数据集**：本次实验使用的数据包含两个常用的图结构数据集：Cora 、Citeseer 。下面分别进行介绍。

   （1）Cora：该数据集是由2708篇机器学习论文作为节点、论文间引用关系作为有向边构成的图数据。具体的数据描述见 https://blog.csdn.net/qq_33254870/article/details/103553661 。数据集下载链接 https://linqs-data.soe.ucsc.edu/public/lbc/cora.tgz 。另外，提供一个数据处理范例链接 https://graphsandnetworks.com/the-cora-dataset/ 。请同学们仔细阅读相关材料，了解文件的具体结构和数据格式。

   （2）Citeseer：该数据集是由3312篇论文及相互引用构成的图数据集。数据集下载链接 https://linqs-data.soe.ucsc.edu/public/lbc/citeseer.tgz 。文件的结构和数据格式与Cora类似。
3. **数据预处理**：你需要通过pytorch或tensorflow所提供的标准数据接口，将原始数据处理为方便模型训练脚本所使用的数据结构，如**torch.utils.data.Dataset/或者使用dgl库进行数据的处理**等。由于这两个数据集是非常常见的公开数据集，你可以参考一些公开代码片段，尤其是github上典型的GCN教程级实现或相关论文的源码。

   `提示：可以使用dgl库/torch_geometric对上述两个数据集进行加载，即无需手动处理数据`
4. **模型搭建**：搭建GCN模型，这一步可以参考网络上公开的源码，但不能直接使用封装过的库**（即不得使用已经封装好的GCN进行构建，请自行实现GCN网络的搭建，如GraphConv需要自己手动实现，不得使用已经封装好的）**。

   `提示：不能直接调用PyG等图网络库，因为本实验的主要目的就是希望大家通过编码熟悉图网络的结构和实现细节。但是允许并鼓励大家多去看看这些库以及其他公开实现的源码，直接改进自己代码是可以的`
5. **模型训练**：将生成的训练集输入搭建好的模型进行前向的 loss 计算和反向的梯度传播，从而训练模型，同时也建议使用网络框架封装的 optimizer 完成参数更新过程。训练过程中记录模型在训练集和验证集上的损失，并绘图可视化。
6. **节点分类**：在两个数据集上按照节点分类任务的需求自行划分训练集、验证集、测试集，并用搭建好的GCN模型进行节点分类。
7. **链路预测**：在两个数据集上按照链路预测任务的需求自行划分训练集、验证集、测试集，并用搭建好的GCN模型进行链路预测。
8. **调参分析**：将训练好的模型在验证集上进行测试，以 **Top 1 Accuracy(ACC)** 作为节点分类指标，**AUC（Area Under the Curve）**作为链路预测任务的指标。然后，对自环、层数、DropEdge 、PairNorm 、激活函数进行调整，再重新训练、测试，并分析对模型性能的影响。
9. **测试性能**：选择你认为最合适的（例如，在验证集上表现最好的）一组超参数，重新训练模型，并在**测试集**上测试（注意，这理应是你的实验中**唯一**一次在测试集上的测试），并记录测试的结果（节点分类：Top1 ACC, 链路预测：AUC）。

实验提交

实验三截止时间：5月12日 23:59:59，线下完成代码检查（关键代码讲解+ 运行展示+结果展示），并需在 bb 系统提交源代码及实验报告，具体要求如下：

1. 全部文件（仅包含.py文件，不需要提交模型文件）打包在一个压缩包内，压缩包命名为：**学号-姓名-exp3.zip。**
2. 代码仅包含.py 文件，请勿包含实验中间结果（例如中间保存的数据集等）,如果有多个代码文件，放在 src/文件夹内。
3. 实验报告提交为.pdf 格式，**包含学号、姓名，内容包括简要的实验过程、关键代码展示**、对**超参数的调试分析**以及测试集上的实验结果。