基于递归神经网络的药物分子ADMET性质预测

Report on 2019.1.4

**潘高翔，蔡丹杨，付思杰**

**北京大学化学与分子工程学院，北京大学工学院**

# 研究背景

略。

# 方法原理

## 2-1 数据预处理

详见README.md文件

## 2-2 输入数据结构

首先利用rdkit包提供的smiles解析器，将以smiles表达式格式存储的分子结构转化为分子图Graph数据结构(自定义无向图)，Graph由节点(Node)构成，Node对象对应原子，原子信息封装在Node中。Node对象还存储了与之连接的其他Node对象引用，即“边”，对应化学键，以及这些引用的权值，即化学键的信息，同样存储于Node中。

Node中，原子信息用125维向量编码，键信息用12维向量编码，除相对原子质量外，其余自然特征全部用one-hot格式编码。具体内容详见model.py的注释部分。(我太懒了，最终报告不能这样)

预测与训练中需应用分子树，可理解为具有根-枝-叶结构的有向无环分子图，同样由Node构成。Graph对象通过最小深度生成树算法(定义在Graph类下的build\_tree方法中)，可建立以任意Node为根节点的分子树，用于递归神经网络模型的预测和训练。

Plus模型中，预测网络除接受递归网络返回的特征外，还接受rdkit计算出的196个分子全局特征，如分子量、氢键受体数、氢键给体数以及一些量子计算特征等。Plus模型一方面增加全局特征，使模型对分子整体关注度上升，预测精度可能因此提高；另一方面，Plus模型也作非Plus模型的对照，以观察递归网络得到的分子特征是否能够完全反映该分子的性质。

## 2-3 递归神经网络

模型输入分子图，输出ADMET预测值。模型可分为性质预测和特征抽取两部分，即：

E为特征抽取函数，返回一个定长特征向量，为单隐层经典神经网络映射，返回预测值。

E函数为求一组递归网络返回值的加和，即：

为递归神经网络映射，为以原子k为根节点的分子树的根节点，为以原子k为根节点的分子树中，k原子节点的返回值向量。

计算当前节点所有子节点在映射下返回值向量，将这些向量拼接上键连信息向量后利用隐层循环采样，并将采样向量求和得到定长向量，与当前节点的原子信息向量拼接，通过单隐层神经网络映射得到返回值向量，数学描述如下：

注意到这是一个递归函数，当递归进行至分子树的叶节点时，由于叶节点无子节点，网络开始逐级向上返回向量值。与Alessandro Lusci等人开发出的递归神经网络原理略有不同：

由于神经网络算法常要求向量长度确定，然而分子树中节点的子节点数目不定，此递归神经网络利用零向量将剩余的特征空间补齐。本文对这一点进行了改进，利用类似于卷积神经网络的方式，对各个子节点循环抽取向量并最终加和，得到定长向量。由于各子节点共享权值，此方法大幅减少了数据维度，且此方法将各子节点放在相同地位上，更加符合实际，也更优雅。

此外，Gilmer等人总结的MPNN方法，以及Kevin Yang等人开发出改进型Directed MPNN，与此处的模型十分相近(建议简要阅读papers文件夹中的MIT\_Analyzing Learned Molecular Representations for Property文献)。但基于MPNN的模型并非通过递归顺序运算，而是通过时序迭代的方式，通过控制迭代次数调节网络输出特征的整体性。这样的作法优点是对于较大规模分子，可控制网络深度而节约计算量，而递归网络的复杂度和深度将会因分子规模变大而爆炸式增长，但MPNN若迭代次数过小，得到的特征将具有强烈的局部特点，不能正确反映分子整体的状况。

# 模型评估

## 2-1 运行环境

Intel® Core™ i5-9300H @CPU 2.40GHz 2.40GHz

Anaconda3, python3.7.4

## 2-2 参数设置：

以下是一代参数

1. 网络(递归内层网络)输入23维(20维+3维键向量)，输出20维，激活函数Tanh；
2. 网络(递归外层网络)输入34维(20维内层输出+14维元素向量)，对于叶节点，输入仅14维，无内层输入，输出20维，激活函数LeakyReLU；
3. 网络(性质预测网络)，输入20维，隐层25维，输出1维，激活函数LeakyReLU，因执行拟合任务，输出无激活函数；
4. Loss(损失函数)，采用MSE损失函数；
5. Optimizer(优化器)，使用Adam算法，学习率0.001，；
6. 训练集共858组数据，测试集286组，占比0.2，训练进行8轮。使用类似自助法采样，训练数据随机。计算机无GPU，训练过程全部在CPU上进行。

二代参数

1. 网络(递归外层网络)输入187维(125维原子信息+12维键信息+50维子节点递归特征输出)，含一个具有150个神经元的隐层，输出50维递归特征，激活函数采用LeakyReLU；
2. SolNet
3. PlusSolNet

## 2-3 评估结果

5折交叉验证训练用时2 hours 50 minutes，交叉验证，测试集上

训练曲线：

等等

## 2-4 讨论分析

从模型设计、表现上分析，于原始文献和两篇前沿文献结果对比

简要介绍HIVNet和PlusHIVNet，着重于对不平衡数据集的处理，如训练时调整损失函数的权重，和可采取的过采样技术，计算复杂度过高，准备用超算了