**Tensorflow训练音频二分类**

**编程软件：**PyCharm；python2.7.13

**系统环境：**Centos7；gcc4.8

**Python主要的模块：**python\_speech\_features；pydub

**1.提取MFCC特征**

**1.1数据集存放结构**

数据一共有498个音频文件存放在audiolib文件夹。正常的音频数据一共有326个，保存在norNoise文件夹，异常的音频数据172个，保存在errNoise文件夹。具体数据存放位置如下：

audiolib

/norNoise

/\*\*\*\*.wav,...,\*\*\*\*.mp3

/errNoise

/\*\*\*\*.wav,…,\*\*\*\*.mp3

**1.2提取MFCC特征**

分别提取正常文件夹与异常文件夹下的音频文件，并作mfcc特征提取，将所有获得到的音频文件mfcc特征，都拼在一起，添加一列正常音频文件代表1，异常音频文件代表0，使用字典将特征保存到本地名称为”hh.pkl”。具体提取mfcc特征的主要步骤如下：

1).使用pydub.AudioSegment模块from\_mp3和from\_wav方法分别读取norNoise文件夹和errNoise文件夹里面的mp3和wav格式的音频文件;

2).使用get\_array\_of\_samples方法获得音频所有数据data，data的大小是1行n列，如果是双声道的数据，data存放多声道的数据方法是[声道1数据，声道2数据，声道1数据，声道2数据，…],总体data的大小是1行n列；

3).使用 python\_speech\_features模块中的mfcc方法，本次采用的是默认的参数，winlen，默认的参数主要有窗长25毫秒，winstep=窗移10毫秒，appendEnergy，默认加入能量列，numcep,倒谱数量13（12维系数+1维能量），nfilt,过滤器26个，nfft数量512，lowfreq,低频0Hz，highfreq,采样率的一半，preemph,预加重的系数0.97**，winfunc,默认不加窗，ceplifter,增加余弦变换系数高频部分的效果，默认为22。**

**2.设置神经网络**

**第一层输入层：128个神经元，**激活函数使用relu函数；

**第二层隐含层：16个神经元，**激活函数使用relu函数；

**第三层输出层：2个神经元，**直接输出。

**更新权值：**梯度下降

**学习率：**0.001

**3.训练方法**

数据的0.5作训练，数据的0.2作验证，数据的0.3作测试。训练阶段，首先对所有的数据进行训练，每运行1000代对训练集进行随机切分1000行输出准确度，把test测试集随机切分成1000行，作训练，直到运行到设置到最大迭代数为止。

最高精度为68.86%。