# 一、前言

语音分类项目，一般目的是对多类别的语音数据进行分类。可以是任意的分类属性，比如 说话人分类，水声分类，鸟声分类，机器声分离。采用交叉熵函数进行有监督的训练。

深度学习训练，遵循下面的基本代码

已知数据样本的集合 S = { x = (data，label) } 神经网络模型 MODEL ，数据集迭代次数 epoch , 单步训练取样本数（batchsize）

For i in range(epoch):

For batch in S:

data，label = batch //(取出一批数据)

pred\_label = model(data) // 一次前向传播

//计算准确率

//计算梯度，反向传播

// 继续循环

使用的样本数据data：

这里统一使用语音的“短时傅里叶谱”或者“对数梅尔频谱（也称为fbank谱）”的频域特征进行训练。其数据的shape为 ：【频率组数，语音 长度】

例如【513，1250】，表示语音的每一帧都有 513组频率，一共有1250帧。

如果你不知道什么叫频谱？请看视频：

<https://www.bilibili.com/video/BV1f3411C7kb/>

对于人声，推荐用 “对数梅尔频谱

对于其他非生命体发出的声音，推荐用 “短时傅里叶谱”

以上两种频谱数据，我们下面都简称“特征数据”。

# 准备python编程环境

请用各种方式准备一个python的编程环境，至少需要有

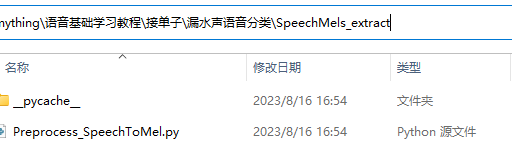
1. cuda版本的pytorch
2. librosa、numpy、matplotlib、pathlib库。

版本都可以是最新版本

# 二、数据特征的提取

训练本语音分类代码，需要提取好你的.wav文件的特征数据。因此，需要先for循环遍历全部的.wav文件，然后将提取的特征数据保存到硬盘里。

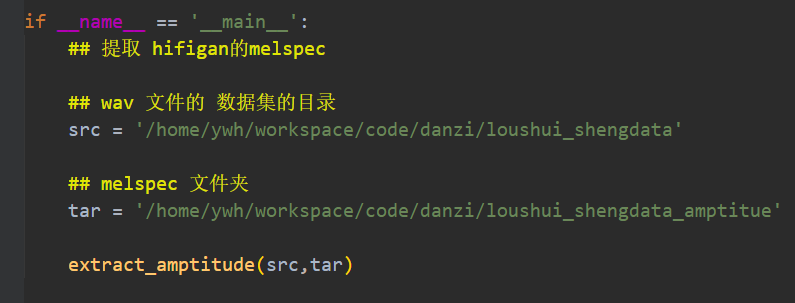
我们在文件



Preprocess\_SpeechToMel.py 这个文件里面，修改你的语音文件夹路径。

Src是你的电脑上的 语音文件夹的路径。

Tar是你希望把特征提取到哪里。 建议就直接复制src，然后加上一个”feat”就好了。

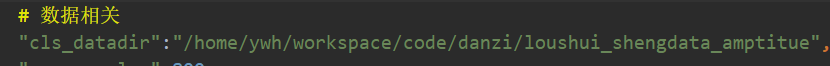


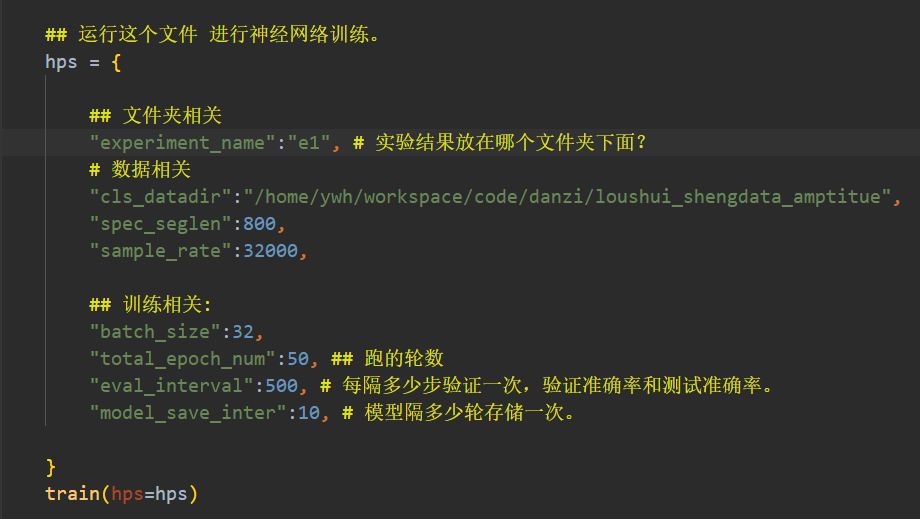
然后运行该文件提取特征。

# 三、运行Train训练

3.1 修改超参数

将 datadir这个参数改成你提取好的特征数据的文件夹。





然后考虑你的 spec\_seglen，一般要根据自己的数据集的情况而定。

比如，你的数据集的语音都是12秒-15秒。而你希望使用10秒钟的语音训练。那么我们需要计算，10秒钟的 等于多少帧？

计算方法如下 ：

10 × 采样率 除以 “hop size” = 训练时的帧数

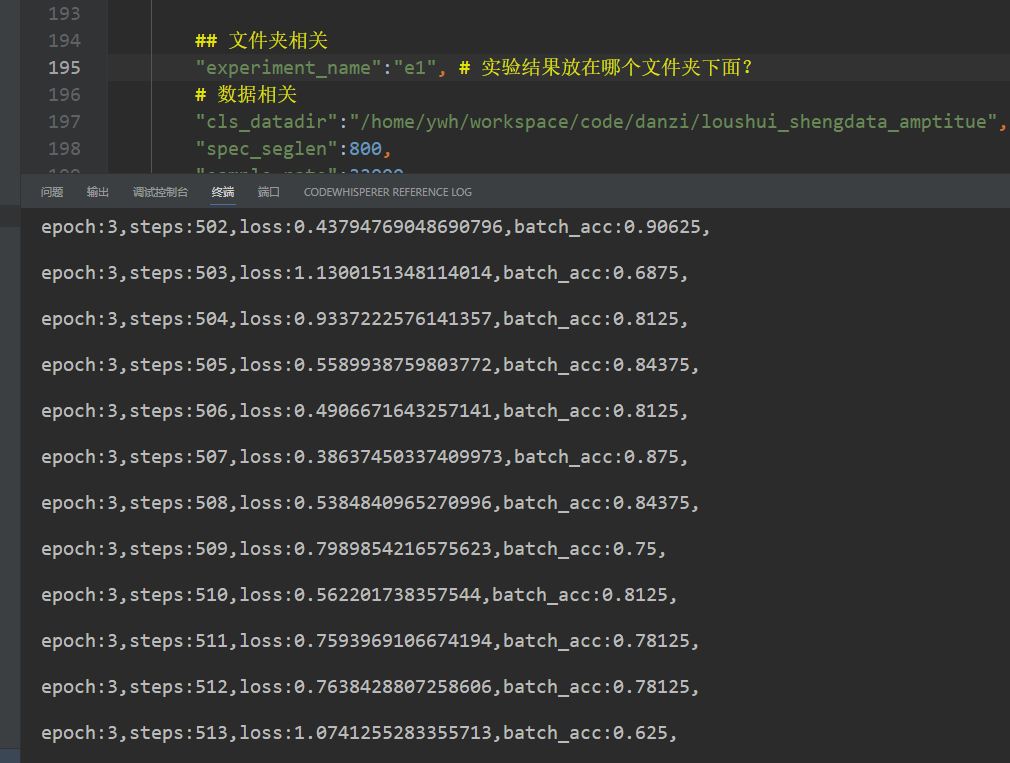
假设，采样率 = 32000， hop size= 256，则训练时的帧数 = 1250 。

我们可以取这个数字，或者一个更小一点的数字。

这个数字取得越大，对显存的占用会越高。但是效果未必提升很大。

3.2 运行Train.py

运行后，看到下面的界面：



即为训练成功。

# 四、结果的保存



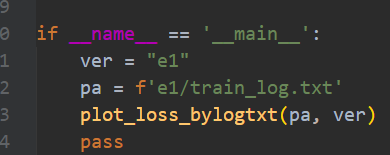
实验结果统一放在这个文件夹下面。你取名为e1，就会产生一个叫e1的文件夹。

4.1 训练准确率、验证准确率、测试准确率

这三个指标都在 e1/ test\_log.txt

4.2 如何画图，观看损失曲线？

在loss\_look.py 文件中，修改你想看的日志文件。比如下面这样



即可生成一堆图片。

