语音指令识别实验报告

181220049 宋磊 人工智能学院 2020 年 11 月 21 日

1 问题描述

问题要求是使用 tensorflow 和 keras 搭建神经网络，完成语音命令词的识别任务，是一个四分类任务。我的思路是首先得到 MFCC 特征和 logmelspectrum 特征，得到 2d 的数据特征，然后利用分类网络 VGG、ResNet 和 DenseNet 完成分类。

2 设计思路

2.1 预先选定的词

我的想法是将发音不易区分的词划分到剩余词中，将他们分为一类，将容易划分的词作为需要识别的语音词。数据集中的 on 和 off 是不容易区分的，因为发音比较相似，所以首先确定将 on 和 off 划入剩余词中，然后需要再选择一个词划分到剩余词中。

因为词的数量很少，所以我选择最后一个剩余词的办法是进行枚举，将 go、left、right、stop 分别和 on、off 一起作为剩余词，然后进行模型的训练，比较在测试集上的准确率。

最终结论是，将 left、right、stop 作为需要识别的词，go、on、off 作为剩余词在验证集和测试集上会同时得到最好的效果。所以实验中这样选用。

2.2 数据预处理

数据预处理主要包括两步，首先对数据进行增强，然后提取 MFCC 特征和 logmelspectrum 特征。

数据增强包括给数据加高斯噪声、time shifting、time stretching、pitch shifting 四种方式，对应代码在 $main.ipynb$ 的数据增强部分，数据增强很重要，在进行数据增强后，过拟合问题减轻了很多。

提取特征的代码在 $tools.py$ 文件的 $preprocess\_mel$ 和 $preprocess\_mfcc$ 函数中，代码使用 librosa 库完成特征提取。

2.3 神经网络模型搭建

我搭建了 vgg、resnet、densenet 三个模型，除了减少了部分层中参数的个数，模型结构和标准模型基本相同。不同模型使用不同的输入，vgg 使用尺寸为 (40, 32, 1) 的 MFCC 特征作为输入，resnet 和 densenet 使用尺寸为 (120, 32, 1) 的 logmelspectrum 特征作为输入。

最终将训练好的三个模型进行集成，集成方式是对每个模型最终预测的 softmax 的结果求和，选择和最大的一个对应的概率。

模型	vgg	resnet	densenet	集成三个模型
准确率	85.47	86.85	85.12	90.66

2.4 过拟合问题

因为给的训练集很少，所以直接训练会产生很严重的过拟合问题。完成实验的过程中，基本一直在解决过拟合问题，我使用了增加验证集、数据增强、dropout、BN、early stop 等方法，缓解过拟合问题，最终提高了在测试集上的准确率。

3 实验结果

最终实验结果如表 [3] 所示。进行模型集成后，效果有了很大提升，我认为原因主要在于不同模型使用了不同的特征输入。