**实验二、语音识别系统搭建及验证**

**一、实验目的**

该实验主要由三个部分组成，分别为GMM的实现、HMM的实现及kaldi

语音识别模型的训练。通过这三个实验初步掌握语音识别的GMM-HMM经典算法，及kaldi工具的使用。

**二、实验环境**

实验平台：Linux

实验1、2：

编程环境：python3

依赖：librosa，numpy，scipy，python\_speech\_features，logging

实验3：

编程环境：shell

依赖：kaldi

**三、实验内容**

实验所在文件夹speech\_experiments/experiments/02-speech-recognition

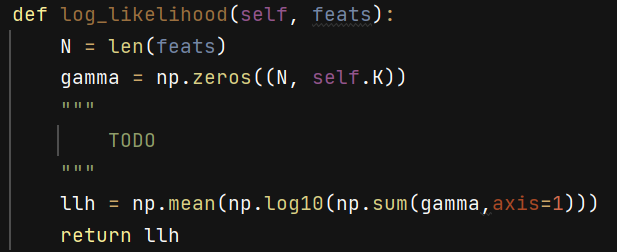
实验二主要包含三个实验，1-GMM实现、2-HMM实现主要基于python，需要补全相关算法的核心代码，运行出结果即可；3-kaldi需要完整运行所给的示例代码得到错误率，选做实验需要根据所给示例的数据准备代码及实验数据集来自己进行数据准备，成功准备4个文件（wav.scp、text、utt2spk、spk2utt），并运行得到错误率即可。

**1.GMM实现（2学时）。**

该实验需要补全EM算法核心代码，进行英语数字语音识别模型的训练及测试（框架已经给出）。

基于13维MFCC特征，每个类别用3个高斯模型建模，目前在测试集上可达95.5%的识别准确率。

（1）补全计算对数似然函数代码。



（2）补全EM算法代码。



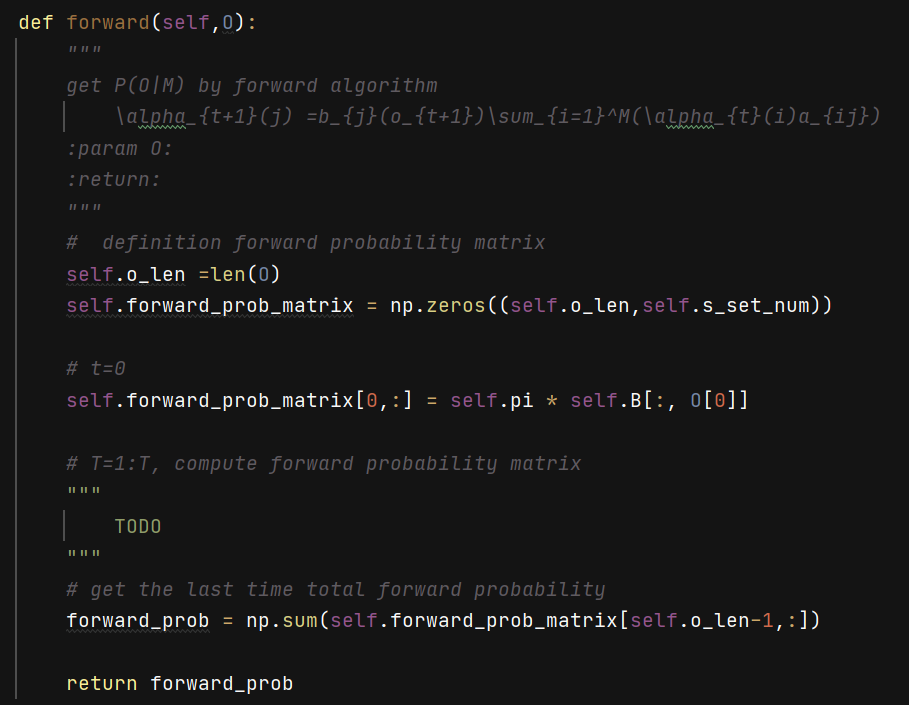
补全代码后运行，实验记录及测试结果会保存在result/gmm.log中。

其他实验说明可见文件夹内的README.md文件。

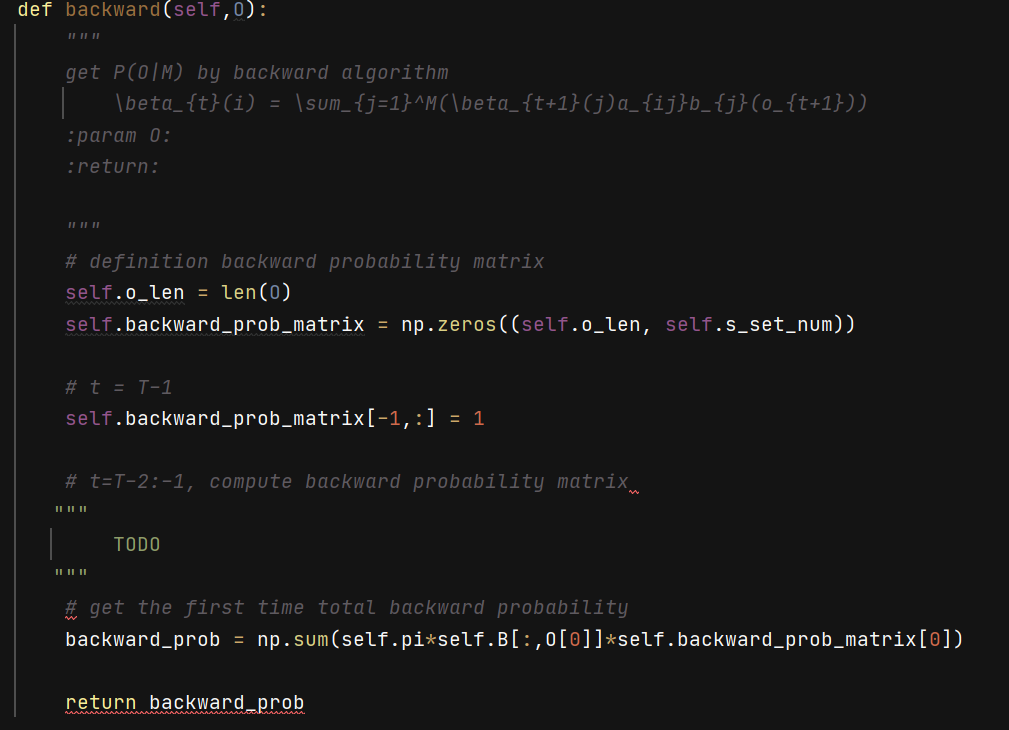
**2. HMM实现（2学时）。**

该实验需要补全HMM的前向算法、后向算法、维特比解码算法的核心代码，从而完成观察序列的路径的解码。

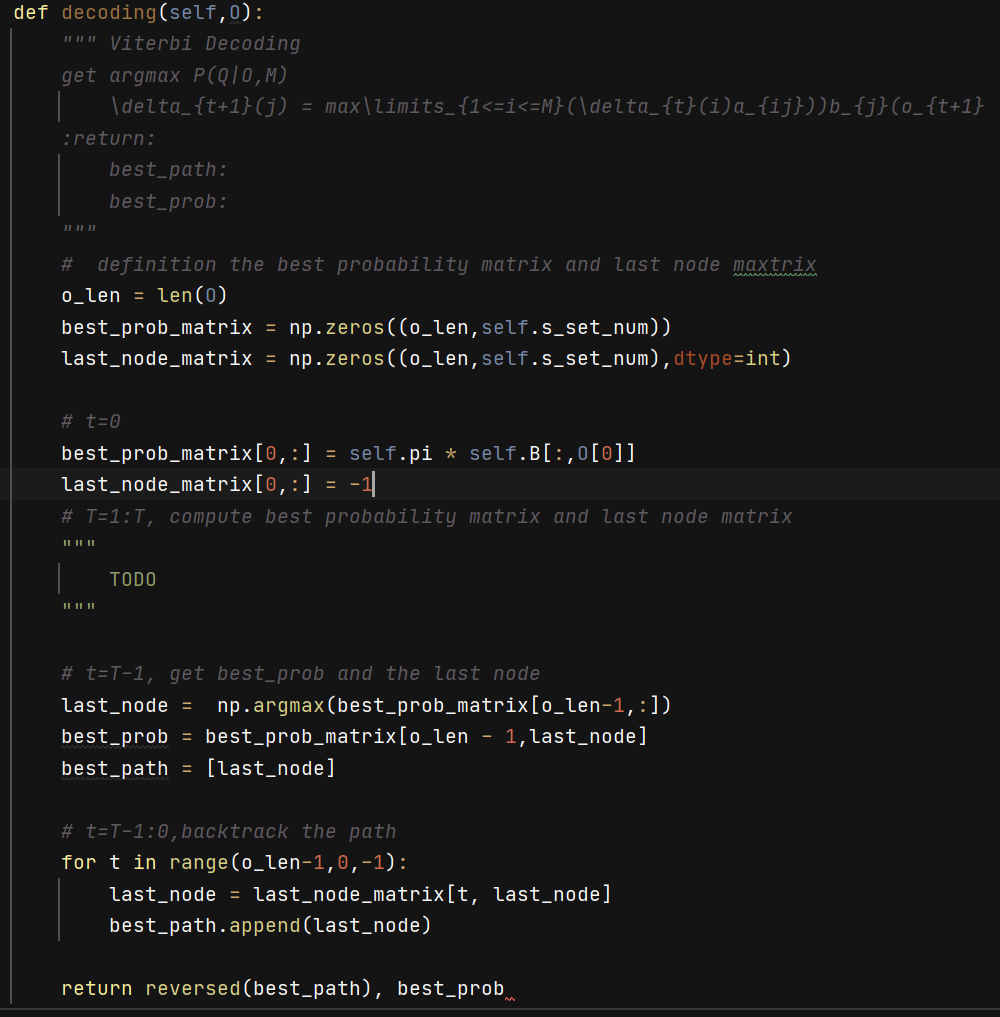
1. 补全前向算法代码。



1. 补全后向算法代码。



（3）补全维特比解码算法代码。



补全代码后运行，实验记录最终预测结果会保存在result/hmm.log中。

其他实验说明可见文件夹内的README.md文件。

**3.** **基于kaldi的连续语音识别系统实现（2学时）**

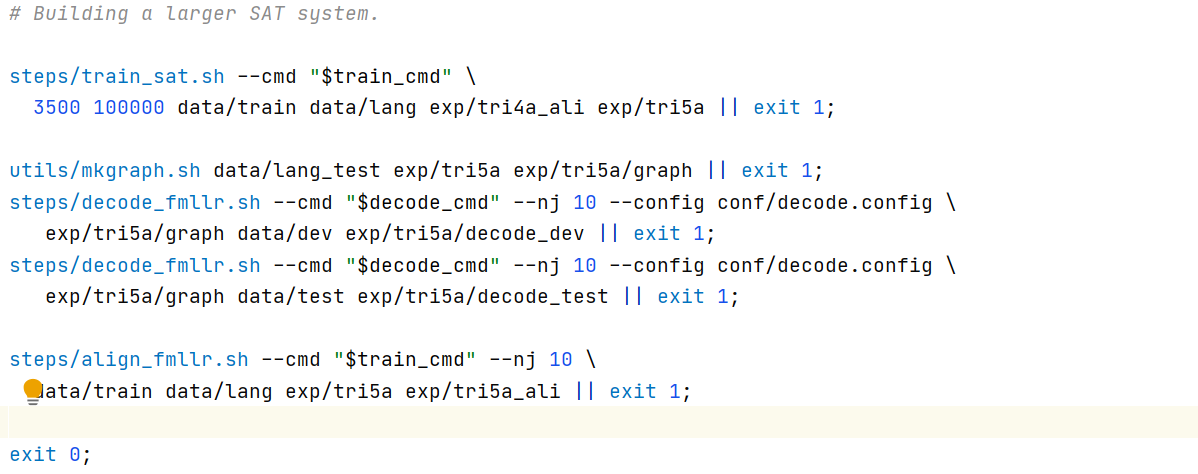
需要使用kaldi进行数据准备和语音识别；

1. 使用kaldi进行识别

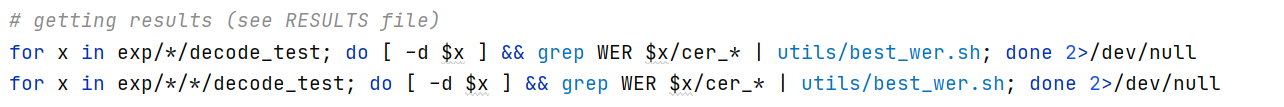
运行所给的数据准备和语音识别shell脚本，进行对所给的数据集进行语音识别实验。







脚本run.sh在kaldi所给示例的s5目录下，最终训练好的声学模型将存为final.mdl，使用下图中的命令进行结果查看。



1. 比较不同声学模型的建模结果

训练单因素声学模型和三音素声学模型，并且使用这两个模型分别对测试集进行解码，对比不同声学模型的解码结果，分析三音素模型和单音素模型的训练过程和运行原理，对解码结果的不同进行分析。

1. 理解实验过，画出完整实验流程图

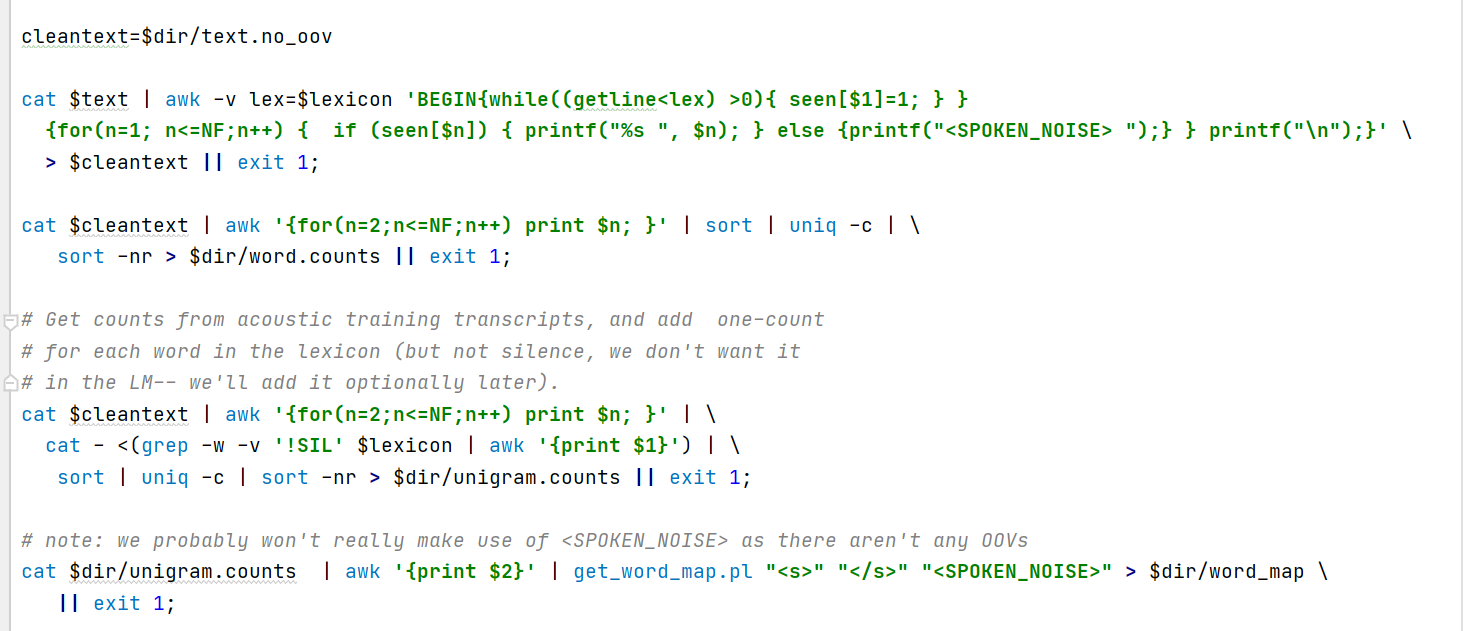
了解kaldi识别过程的运行机制，画出kaldi语音识别实验完整的搭建流程图，包括数据准备，特征提取，声学模型训练过程，语言模型训练过程，解码过程等。并且对每个步骤进行分析，写出每个步骤的作用和意义。

使用kalid进行语音识别实验，理解画流程图总结实验流程

**4.基于kaldi的连续语音识别验证实验（选做）**

1. 更改语言模型进行实验

Kaldi对所给数据进行训练语言模型的脚本为aishell\_train\_lms.sh,在脚本中更改训练语言模型所使用的文本，已给出的文本中使用了训练集的文本当作word.counts生成的来源，并且与lexicon共同生成的word\_map进行语言模型的训练，而不同来源的文本对模型的识别有着不同的影响，将原始输入的train/text进行更改，更改为所给出的text1，text2，以及自己准备text3，并且修改脚本生成不同的语言模型，进行kaldi语音识别实验，并对实验结果进行分析。



如图所示，修改cleantext的文件来源，修改代码，最后生成不同的语言模型，进行后续实验。

**四、结果验收**

**实验1：**

（1） 补全完成后的代码。

（2） result文件夹中的日志文件gmm.log。

**实验2：**

1. 补全完成后的代码。
2. Result文件夹中的日志文件hmm.log。

**实验3:**

1. 运行完成截图及错误率文件。
2. 单因素和三音素建模后的解码结果，及其分析文档。
3. 对于整个kaldi语音识别实验运行过程的详细分析文档（需要流程图）。

**实验4:**

1. 数据准备shell脚本。
2. 不同语语言模型进行语音识别的运行结果，以及文档形式分析报告。
3. 不同语言模型的错误率文件。

**五、参考资料**  
**实验1：**

[1] [nwpuaslp/ASR\_Course [github]]

(https://github.com/nwpuaslp/ASR\_Course/tree/master/03-GMM-EM)   
[2] [GMM\EM算法详解——附0-9孤立词识别demo(python实现) [CSDN]]

(https://blog.csdn.net/weixin\_39529413/article/details/117200165)   
[3] [stober/gmm [github]]

(<https://github.com/stober/gmm/blob/master/src/gmm.py>)

**实验2：**

[1] [Eilene/HMM-python [github]]

(https://github.com/Eilene/HMMpython/blob/master/HMM.py)   
[2] [nwpuaslp/ASR\_Course [github]]

(https://github.com/nwpuaslp/ASR\_Course/tree/master/04-HMM)

**实验3、4：**

[1] [端到端(E2E)语音识别]([端到端(E2E)语音识别 (xmu.edu.cn)](https://speech.xmu.edu.cn/_upload/article/files/bb/6c/02a02295420294f6cee55184f035/1a87cbee-7beb-47a7-9e97-f538d27ddf21.pdf))

[2] [kaldi-train\_lm.sh]([kaldi-train\_lm.sh - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/82333064))

[3] [kaldi学习笔记 -- 构造语言模型相关脚本]( [kaldi学习笔记 -- 构造语言模型相关脚本 -- ami\_train\_lms.sh，utils/format\_lm.sh\_DuishengChen的专栏-CSDN博客](https://blog.csdn.net/DuishengChen/article/details/52514477))

[4] [音频工具kaldi部署及模型制作调研学习]([音频工具kaldi部署及模型制作调研学习 - 努力跑在岁月之前 - 博客园 (cnblogs.com)](https://www.cnblogs.com/relax-zw/p/11972617.html))