



链滴

kaldi 在 mac 下的初体验

作者: [flowaters](#)

原文链接: <https://ld246.com/article/1547886299887>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

背景

Kaldi是使用人数最多的语音识别开源工具，而且在不断的更新[2]。

更多的背景介绍见[2]，本文尝试编译Kaldi并且跑通一些小的例子。

源码编译

下载

```
git clone https://github.com/kaldi-asr/kaldi
```

编译tools

参考tools/INSTALL文件

安装依赖包

```
brew install automake autoconf python3
```

检查下依赖情况

```
[note@abeffect tools]$ sh extras/check_dependencies.sh  
extras/check_dependencies.sh: all OK.
```

编译

```
make
```

结果

```
Warning: IRSTLM is not installed by default anymore. If you need IRSTLM  
Warning: use the script extras/install_irstlm.sh  
All done OK.
```

安装扩展

在extras目录下有多个扩展，可以选择性的安装。

编译src

参考src/INSTALL文件

```
cd src  
./configure --shared  
make depend -j 8  
make -j 8
```

结果

echo Done
Done

可执行文件见latbin目录

```
[note@abeffect src]$ ls latbin/ | grep -v cc$ | grep -v o$  
Makefile  
lattice-1best  
lattice-add-penalty  
lattice-add-trans-probs  
lattice-align-phones  
lattice-align-words  
lattice-align-words-lexicon  
lattice-arc-post  
lattice-best-path  
lattice-boost-ali  
lattice-combine  
lattice-compose  
lattice-confidence  
lattice-copy  
lattice-copy-backoff  
lattice-depth  
lattice-depth-per-frame  
lattice-determinize  
lattice-determinize-non-compact  
lattice-determinize-phone-pruned  
lattice-determinize-phone-pruned-parallel  
lattice-determinize-pruned  
lattice-determinize-pruned-parallel  
lattice-difference  
lattice-equivalent  
lattice-expand-ngram  
lattice-interp  
lattice-limit-depth  
lattice-lmrescore  
lattice-lmrescore-const-arpa  
lattice-lmrescore-kaldi-rnnlm  
lattice-lmrescore-kaldi-rnnlm-pruned  
lattice-lmrescore-pruned  
lattice-lmrescore-rnnlm  
lattice-mbr-decode  
lattice-minimize  
lattice-oracle  
lattice-project  
lattice-prune  
lattice-push  
lattice-rescore-mapped  
lattice-rmali  
lattice-scale  
lattice-to-ctm-conf  
lattice-to-fst  
lattice-to-mpe-post  
lattice-to-nbest  
lattice-to-phone-lattice
```

```
lattice-to-post
lattice-to-smbr-post
lattice-union
linear-to-nbest
nbest-to-ctm
nbest-to-lattice
nbest-to-linear
nbest-to-prons
```

使用

原来voxforge例子中的语音库需要12.6G的空间，需要预留20G的空间来做实验[6]。

所以还是先从yesno来入门吧。

yesno

运行

```
[note@abeffect kaldi]$ cd egs/yesno/s5/
[note@abeffect s5]$ ./run.sh
```

结果

```
.....
steps/diagnostic/analyze_lats.sh: see stats in exp/mono0a/decode_test_yesno/log/analyze_latt
ce_depth_stats.log
local/score.sh --cmd utils/run.pl data/test_yesno exp/mono0a/graph_tgpr exp/mono0a/deco
e_test_yesno
local/score.sh: scoring with word insertion penalty=0.0,0.5,1.0
%WER 0.00 [ 0 / 232, 0 in , 0 del, 0 ub ] exp/mono0a/decode_t_e t_ ye no/wer_10_1.0
```

可视化

确保安装了graphviz

```
brew install graphviz
```

语言模型可视化

```
.././../tools/openfst-1.6.5/bin/fstprint ./data/lang_test_tg/G.fst
0 0 2 2 2.30258512
0 0 3 3 2.30258512
0 2.30258512
```

```
.././../tools/openfst-1.6.5/bin/fstdraw ./data/lang_test_tg/G.fst | dot -T ps > g.ps
```

词典文件可视化

L.fst文件

```
$ ../../tools/openfst-1.6.5/bin/fstprint ./data/lang_test_tg/L.fst
```

```
0 1 0 0 0.693147182
0 1 1 0 0.693147182
1 1 1 1
1 1 3 2 0.693147182
1 2 3 2 0.693147182
1 1 2 3 0.693147182
1 2 2 3 0.693147182
1
2 1 1 0
```

```
../../tools/openfst-1.6.5/bin/fstdraw ./data/lang_test_tg/L.fst | dot -T ps > 1.ps
```

L_disambig文件

```
$ ../../tools/openfst-1.6.5/bin/fstprint ./data/lang_test_tg/L_disambig.fst
```

```
0 1 0 0 0.693147182
0 2 1 0 0.693147182
1 1 1 1
1 1 3 2 0.693147182
1 3 3 2 0.693147182
1 1 2 3 0.693147182
1 3 2 3 0.693147182
1 1 4 4
1
2 1 5 0
3 2 1 0
```

timit

没有下载到完整的timit语料，放弃。。

原理

简单补充一下 yesno 的原理。

原始音频

共60个wav文件，如waves_yesno/0_0_0_0_1_1_1_1.wav，内容为yes, no组成。

参考

1. [Kaldi Speech Recognition Toolkit: DFSMN](#)
2. [从声学模型算法总结 2016 年语音识别的重大进步 | 硬创公开课](#)
3. [LibriSpeech language models, vocabulary and G2P models](#)
4. [LibriSpeech ASR corpus](#)
5. [【资源】最好用的 AI 开源数据集 Top 39: 计算机视觉、NLP、语音等 6 大类](#)
6. [语音识别系统之kaldi-----voxforge实例](#)
7. [语音识别系统kaldi----实例说明](#)

8. [语音识别工具箱之kaldi介绍](#)
9. [kaldi中FST的可视化-以yesno为例](#)
10. [kaldi yesno example](#)
11. [yesno孤立词识别kaldi脚本](#)
12. [Kaldi-yesno详解](#)
13. [语音识别及处理](#)