

基于kaldi的thchs30训练实践V1.4

6-24-2019_slip_v1.0

6-25-2019_slip_v1.1

6-26-2019_slip_v1.2

6-28-2019_slip_v1.3

7-1-2019_slip_v1.4

首先在老师给的机器上看一下，到/voice_rec/kaldi/egs/thchs30/s5下，不出所料没有data文件，但是有一个download_and_untar.sh，看了一下代码太多，所以还是直接自己动手丰衣足食。

```

model@student:~/voice_rec/kaldi/egs/thchs30/s5/local$ cat download_and_untar.sh
#!/bin/bash

# Copyright 2014 Johns Hopkins University (author: Daniel Povey)
# Copyright 2016 Tsinghua University (author: Dong Wang)
# Apache 2.0

# Adapted from librispeech recipe local/download_and_untar.sh

remove_archive=false

if [ "$1" == --remove-archive ]; then
    remove_archive=true
    shift
fi

if [ $# -ne 3 ]; then
    echo "Usage: $0 [--remove-archive] <data-base> <url-base> <corpus-part>"
    echo "e.g.: $0 /nfs/public/materials/data/thchs30-openslr www.openslr.org/resources/18 data_thchs30"
    echo "With --remove-archive it will remove the archive after successfully un-tarring it."
    echo "<corpus-part> can be one of: data_thchs30, test-noise, resource"
fi

data=$1
url=$2
part=$3

if [ ! -d "$data" ]; then
    echo "$0: no such directory $data"
    exit 1;
fi

part_ok=false
list="data thchs30 test-noise resource"
for x in $list; do
    if [ "$part" == $x ]; then part_ok=true; fi
done
if ! $part_ok; then
    echo "$0: expected <corpus-part> to be one of $list, but got '$part'"
    exit 1;
fi

if [ -z "$url" ]; then
    echo "$0: empty URL base."
    exit 1;
fi

if [ -f $data/$part/.complete ]; then
    echo "$0: data part $part was already successfully extracted, nothing to do."
    exit 0;
fi

sizes="6453425169 1971460210 24813708"

if [ -f $data/$part.tgz ]; then
    size=$(/bin/ls -l $data/$part.tgz | awk '{print $5}')
    size_ok=false
    for s in $sizes; do if [ $s == $size ]; then size_ok=true; fi; done
    if ! $size_ok; then
        echo "$0: removing existing file $data/$part.tgz because its size in bytes $size"

```

download_and_untar.sh

下载的处理过程:

```

wget http://cn-mirror.openslr.org/resources/18/data_thchs30.tgz
wget http://cn-mirror.openslr.org/resources/18/test-noise.tgz
wget http://cn-mirror.openslr.org/resources/18/resource.tgz

```

完成后mkdir thchs30-openslr并解压到./s5/thchs30-openslr。

修改./s5/cmd.sh为:

```
#export train_cmd=queue.pl
#export decode_cmd="queue.pl --mem 4G"
#export mkgraph_cmd="queue.pl --mem 8G"
#export cuda_cmd="queue.pl --gpu 1"
export train_cmd=run.pl
export decode_cmd="run.pl --mem 4G"
export mkgraph_cnd="run.pl --mem 8G"

export cuda_cmd="run.pl --gpu 1"
```

修改./s5/run.sh为:

```
#n=8      #parallel jobs
n=4      #change by num of cpuCores
#thchs=/nfs/public/materials/data/thchs30-openslr

thchs=/home/model/voice_rec/kaldi/egs/thchs30/s5/thchs30-openslr
```

bash run.sh 便开始训练了。它大概有几个过程: 数据准备, monophone单音素训练, tri1三因素训练, trib2进行lda_mllt特征变换, trib3进行sat自然语言适应, trib4做quick, 后面就是dnn了。

```
model@student:~/voice_rec/kaldi/egs/thchs30/s5$ bash run.sh
creating data/{train,dev,test}
cleaning data/train
preparing scp and text in data/train
```

先到这里, 让他跑着。

今天一早上去一看, 发现并没有进程。

```
Last login: Mon Jun 24 14:24:34 2019 from 211.87.229.29
model@student:~$ ps
  PID TTY          TIME CMD
 7379 pts/2        00:00:00 bash
 7393 pts/2        00:00:00 ps
model@student:~$ jobs
model@student:~$
```

因为如果成功运行结束, thchs30/s5/exp中会有变化, 我们可以进去看一下:

```
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tril$ ls | grep f
final.mdl
final.occs
fst.1.gz
fst.2.gz
fst.3.gz
fst.4.gz
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tril$ cd graph_word/
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tril/graph_word$ ls
disambig_tid.int  HCLG.fst  num_pdfs  phones  phones.txt  words.txt
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tril/graph_word$
```

tri1/final.mdl即为输出的模型, 此外graph_word文件夹下面有words.txt和HCLG.fst, 一个是字典, 一个是有限状态机。

然后在kaldi/src下

```
make ext
```

编译扩展程序。经过几分钟的编译后, 可以在src/onlinebin下看到

```

model@student:~/voice_rec/kaldi/src/onlinebin$ ls
java-online-audio-client      online-audio-server-decode-faster.o  online-server-gmm-decode-faster
Makefile                     online-gmm-decode-faster             online-server-gmm-decode-faster.cc
online-audio-client          online-gmm-decode-faster.cc          online-server-gmm-decode-faster.o
online-audio-client.cc       online-gmm-decode-faster.o           online-wav-gmm-decode-faster
online-audio-client.o        online-net-client                    online-wav-gmm-decode-faster.cc
online-audio-server-decode-faster  online-net-client.cc                online-wav-gmm-decode-faster.o
online-audio-server-decode-faster.cc  online-net-client.o
model@student:~/voice_rec/kaldi/src/onlinebin$

```

online-wav-gmm-decode-faster 用来回放wav文件来识别的，online-gmm-decode-faster用来从麦克风输入声音来识别的。

现在配置一个demo：

```

cd egs/voxforge/
cp -r ./online_demo/ ../thchs30/#将voxforge下的online_demo cp 到thchs30下
cd ../thchs30/
cd online_demo/
mkdir online-data#创建两个目录
mkdir work
cd online-data/
mkdir audio#创建两个目录
mkdir models
cd models/
mkdir tri1#在models下创建tri1
cd tri1/
cp ../../../../s5/exp/tri1/35.mdl ../#将thchs30/s5/exp/tri1下的两个文件 cp 到当前目录
cp ../../../../s5/exp/tri1/final.mdl ../
cp ../../../../s5/exp/tri1/graph_word/words.txt ../#将./graph_word下的两个文件 cp 到当前目录
cp ../../../../s5/exp/tri1/graph_word/HCLG.fst ../

```

修改后的文件目录应该是这样的：

来自：
https://blog.csdn.net/m0_38055352/article/details/82560600tdsourcetag=s_pcqq_aio
 msg

```

online_demo
├── online-data
│   ├── audio
│   │   ├── 1.wav
│   │   ├── 2.wav
│   │   ├── 3.wav
│   │   ├── 4.wav
│   │   ├── 5.wav
│   │   └── trans.txt
│   └── models
│       └── tri1
│           ├── 35.mdl
│           ├── final.mdl
│           ├── HCLG.fst
│           └── words.txt
├── README.txt
├── run.sh
└── work[这个文件夹运行run.sh成功后才会出现]
    ├── ali.txt
    ├── hyp.txt
    └── input.scp

```

```
|— ref.txt
|— trans.txt
```

修改thchs30/online_demo/run.sh:

```
:'#Here is changed by slip,we donot need online data
if [ ! -s ${data_file}.tar.bz2 ]; then
    echo "Downloading test models and data ..."
    wget -T 10 -t 3 $data_url;
    if [ ! -s ${data_file}.tar.bz2 ]; then
        echo "Download of $data_file has failed!"
        exit 1
    fi
fi
'
```

```
ac_model_type=tri1
#changed by slip,to be our url
```

```
online-wav-gmm-decode-faster --verbose=1 --rt-min=0.8 --rt-max=0.85\
--max-active=4000 --beam=12.0 --acoustic-scale=0.0769 \
scp:$decode_dir/input.scp $ac_model/final.mdl $ac_model/HCLG.fst \
$ac_model/words.txt '1:2:3:4:5' ark,t:$decode_dir/trans.txt \
ark,t:$decode_dir/ali.txt $trans_matrix;;
#changed by slip,from model into final.mdl
```

现在便可以开始run了:

```
./run.sh      #开始回放识别, 即识别.wav文件
./run.sh -test-mode live    #从麦克风识别
```

将B6_390至B6_395共6个文件cp到audio目录下, 并且./run.sh:

```

model@student:~/voice_rec/kaldi/egs/thchs30/online_demo$ ./run.sh
./run.sh: line 42: $':#Here is changed by slip,we donot need online data\nif [ ! -s ${data_file}.tar.bz2 ]; then\n
echo "Downloading test models and data ..."\n      wget -T 10 -t 3 $data_url;\n\n      if [ ! -s ${data_file}.tar.bz2 ]; t
hen\n          echo "Download of $data_file has failed!"\n          exit 1\n      fi\nfi\n': command not found

SIMULATED ONLINE DECODING - pre-recorded audio is used

The (bigram) language model used to build the decoding graph was
estimated on an audio book's text. The text in question is
"King Solomon's Mines" (http://www.gutenberg.org/ebooks/2166).
The audio chunks to be decoded were taken from the audio book read
by John Nicholson(http://librivox.org/king-solomons-mines-by-haggard/)

NOTE: Using utterances from the book, on which the LM was estimated
      is considered to be "Cheating" and we are doing this only for
      the purposes of the demo.

You can type "./run.sh --test-mode live" to try it using your
own voice!

online-wav-gmm-decode-faster --verbose=1 --rt-min=0.8 --rt-max=0.85 --max-active=4000 --beam=12.0 --acoustic-scale=0.0
769 scp:./work/input.scp online-data/models/tril/final.mdl online-data/models/tril/HCLG.fst online-data/models/tril/wo
rds.txt 1:2:3:4:5 ark,t:./work/trans.txt ark,t:./work/ali.txt
File: B6 390
在 列宁 并没有 因此 而 否定 托尔斯泰 反而 称赞 他 的 作品 是 俄国 革命 的 一面 镜

File: B6 391
许多 学者 认为 都 荒漠 高 哭诉 保存 的 体育 文物 的 古 希腊 和 罗马 的 可能 早 更 完美

File: B6 392
你 不 去 我 外侧 有 个 县城 新郎 涅 不 必 担心 我 激励 恢复 的 时候 是否 会 不在 而已

File: B6 393
在 国会 一些 议员 也 已 提出 议案 要求 日本 在 虎 殿 内 完全 消灭 对 美 贸易 逆差

File: B6 394
合理 的 爱 与 和 别 的 人 怜爱 对于 他们 说 有 什么 区别

一 撇 撇嘴 恶毒 的 这样 想

File: B6 395
郑 早日 为 扫荡 的 国民党 文具 还 得 起 泰 人 与 我 联系 要求 涉及 部队 转移 当 我 淮安 根据地

./run.sh: line 103: online-data/audio/trans.txt: No such file or directory
compute-wer --mode=present ark,t:./work/ref.txt ark,t:./work/hyp.txt
WARNING (compute-wer[5.5.388~1-777f8]):Open():util/kaldi-table-inl.h:513) Failed to open stream ./work/ref.txt
ERROR (compute-wer[5.5.388~1-777f8]):SequentialTableReader():util/kaldi-table-inl.h:860) Error constructing TableReader
: rspecifier is ark,t:./work/ref.txt

[ Stack-Trace: ]
/home/model/voice_rec/kaldi/src/lib/libkaldi-base.so(kaldi::MessageLogger::LogMessage() const+0x82c) [0x7f0a6070d2ca]
compute-wer(kaldi::MessageLogger::LogAndThrow::operator=(kaldi::MessageLogger const&)+0x21) [0x40b89d]
compute-wer(kaldi::SequentialTableReader<kaldi::TokenVectorHolder>::SequentialTableReader(std::::__cxx11::basic_string<<
har, std::char_traits<char>, std::allocator<char> > const&)+0xal) [0x40eal7]
compute-wer(main+0x2ad) [0x40a623]
/lib/x86_64-linux-gnu/libc.so.6(_libc_start_main+0xf0) [0x7f0a5fba8830]
compute-wer( start+0x29) [0x40a2a9]

```

可以看到虽然错误率有点高，但是还是基本跑通了。

由于前一天第一次训练，没有发现/best_wer 文件，所以又训练了一遍，这次在 thchs30/s5/exp/tri1/decode_test_word/scoring_kaldi 下发现了该文件，打开：

```

model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tri1/decode_test_word/scoring
_kaldi$ cat best_wer
%WER 36.23 [ 29400 / 81139, 525 ins, 1080 del, 27795 sub ] exp/tri1/decode_test_
word/wer_10_0.0

```

我们可以看到，错误率在36.23%。

因为突然关机，之前写的文档进度没保存，可见及时存档的重要性。。。下面补充一下：

在测试完数据之后，我们准备采取我们自己真实的声音进行识别，看一下识别效率，作为对照，我们准备了两组数据：取自数据集的文本，任意取自互联网的文本。

A19_123.wav.trn

另外 假设 不 麻烦 请 关照 一下 内务 及 星空 花园 原则上 别 让 屋子 变成 鬼屋 就好了 啦 云云

A19_124.wav.trn

另外 女单 中国队 还有 韩晶 娜 和 尧 燕 奥运 排名 第六 七位 也 可 与 高手 一 搏

A19_125.wav.trn

工厂 和 厂房 依山 而 建 全部 配备 排污 系统 你 见 不到 黑 烟 听不到 噪声 也 看不到 污水

A19_126.wav.trn

对於 她 在 人 赃 俱 获 的 情况 下 仍 强 辞 夺 理 颇 感 有趣 她 似乎 不知道 绝望 为 何物

A19_127.wav.trn

要想扶优汰劣首先要解决谁优谁劣的问题要判定谁优谁劣必须得有一个衡量的尺子

A19_128.wav.trn

仅绘画而论齐白石是巍巍昆仑可这位附庸风雅的门外汉连一块石头都不是

A19_129.wav.trn

他患有风湿性腰疼病一粘潮湿劳累就疼痛难忍但装井口又必须弯腰弓背钻进满是泥水的钻台下边干活

A19_130.wav.trn

大花鞋的殷勤与自信早已烟消云散她抱着双臂冷冷地看着这一切

A19_131.wav.trn

去年二月未满十岁的徐敏又被上海前进业余进修学院录取继续学习新概念英语第四册

A19_132.wav.trn

得到李公朴噩耗闻一多怒愤填膺拍案而起怒斥反动派卑鄙无耻

以上为取自数据集的文本

kaldi是支持麦克风传入实时识别的，但是由于没有麦克风装置，所以在经过同学录音，然后传入识别后，结果如下：

铺天盖地的各种消息是自由球员市场即将开启的重要标志。
不过看似杂乱无序、纠缠成团的局面，却有一个被视为重中之重的线头。
只要沿着这条线抽丝剥茧，所有问题都将明朗化。
手术事宜要待医生对其进行进一步的检查、治疗和评估后再确定。
把这份爱延续下去，这将会是女儿一生中最宝贵的财富。
燕子去了，有再来的时候；杨柳枯了，有再青的时候；桃花谢了，有再开的时候。
盼望着，盼望着，东风来了，春天的脚步近了。
现代散文家朱自清的白话散文对“五四”以后的散文作家产生过一定的影响。
母亲在牌桌上遇见一位太太，她有个女儿，透着聪明伶俐。
随着各地公积金管理政策的优化和完善，不少地方简化了公积金提取手续

以上为取自网络的文本

在经过同学录音，然后传入识别后，结果如下：

在之前的训练中，由于服务器没有GPU，所以没有进行DNN的训练。

```
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tri4b_dnn/log$ cat train_nnet.log
# steps/nnet/train.sh --copy_feats false --cmvn-opts "--norm-means=true --norm-vars=false" --hid-layers 4 --hid-dim 1024 --learn-rate 0.008 data/fbank/train data/fbank/dev data/lang exp/tri4b_ali exp/tri4b_ali_cv exp/tri4b_dnn
# Started at Tue Jun 25 11:49:02 CST 2019
#
steps/nnet/train.sh --copy_feats false --cmvn-opts --norm-means=true --norm-vars=false --hid-layers 4 --hid-dim 1024 --learn-rate 0.008 data/fbank/train data/fbank/dev data/lang exp/tri4b_ali exp/tri4b_ali_cv exp/tri4b_dnn

# INFO
steps/nnet/train.sh : Training Neural Network
dir : exp/tri4b_dnn
Train-set : data/fbank/train 10000, exp/tri4b_ali
CV-set : data/fbank/dev 893 exp/tri4b_ali_cv

LOG ([5.5.388~1-777f8]:main():cuda-gpu-available.cc:60)

## IS CUDA GPU AVAILABLE? 'student' ###
## CUDA WAS NOT COMPILED IN! ###
To support CUDA, you must run 'configure' on a machine that has the CUDA compiler 'nvcc' available.
# Accounting: time=0 threads=1
# Ended (code 1) at Tue Jun 25 11:49:02 CST 2019, elapsed time 0 seconds
model@student:~/voice_rec/kaldi/egs/thchs30/s5/exp/tri4b_dnn/log$
```

在获得GPU服务器后，由于权限不足，又没能安装kaldi需要的依赖，亦不能完成DNN，所以我们准备用在基础的服务器上用CPU跑一下DNN：

```

model@student:~/voice_rec/kaldi/egs/thchs30/s5$ bash run.sh --skip-cuda-check true
creating data/{train,dev,test}
cleaning data/train
preparing scps and text in data/train
cleaning data/dev
preparing scps and text in data/dev
cleaning data/test
preparing scps and text in data/test
creating test_phone for phone decoding
steps/make_mfcc.sh --nj 4 --cmd run.pl data/mfcc/train exp/make_mfcc/train mfcc/train
utils/validate_data_dir.sh: Successfully validated data-directory data/mfcc/train
steps/make_mfcc.sh: [info]: no segments file exists: assuming wav.scp indexed by utterance.
steps/make_mfcc.sh: Succeeded creating MFCC features for train
steps/compute_cmvn_stats.sh data/mfcc/train exp/mfcc_cmvn/train mfcc/train
Succeeded creating CMVN stats for train
steps/make_mfcc.sh --nj 4 --cmd run.pl data/mfcc/dev exp/make_mfcc/dev mfcc/dev
utils/validate_data_dir.sh: Successfully validated data-directory data/mfcc/dev
steps/make_mfcc.sh: [info]: no segments file exists: assuming wav.scp indexed by utterance.
steps/make_mfcc.sh: Succeeded creating MFCC features for dev
steps/compute_cmvn_stats.sh data/mfcc/dev exp/mfcc_cmvn/dev mfcc/dev
Succeeded creating CMVN stats for dev
steps/make_mfcc.sh --nj 4 --cmd run.pl data/mfcc/test exp/make_mfcc/test mfcc/test
utils/validate_data_dir.sh: Successfully validated data-directory data/mfcc/test
steps/make_mfcc.sh: [info]: no segments file exists: assuming wav.scp indexed by utterance.
steps/make_mfcc.sh: Succeeded creating MFCC features for test
steps/compute_cmvn_stats.sh data/mfcc/test exp/mfcc_cmvn/test mfcc/test
Succeeded creating CMVN stats for test
make word graph ...
utils/prepare_lang.sh --position_dependent_phones false data/dict <SPOKEN_NOISE> data/local/lang data/lang
Checking data/dict/silence_phones.txt ...
--> reading data/dict/silence_phones.txt
--> text seems to be UTF-8 or ASCII, checking whitespaces
--> text contains only allowed whitespaces
--> data/dict/silence_phones.txt is OK

Checking data/dict/optional_silence.txt ...
--> reading data/dict/optional_silence.txt
--> text seems to be UTF-8 or ASCII, checking whitespaces
--> text contains only allowed whitespaces
--> data/dict/optional_silence.txt is OK

Checking data/dict/nonsilence_phones.txt ...
--> reading data/dict/nonsilence_phones.txt
--> text seems to be UTF-8 or ASCII, checking whitespaces
--> text contains only allowed whitespaces
--> data/dict/nonsilence_phones.txt is OK

Checking disjoint: silence_phones.txt, nonsilence_phones.txt
--> disjoint property is OK.

Checking data/dict/lexicon.txt
--> reading data/dict/lexicon.txt
--> text seems to be UTF-8 or ASCII, checking whitespaces
--> text contains only allowed whitespaces
--> data/dict/lexicon.txt is OK

Checking data/dict/lexiconp.txt
--> reading data/dict/lexiconp.txt
--> text seems to be UTF-8 or ASCII, checking whitespaces

```

感谢:

https://blog.csdn.net/m0_38055352/article/details/82560600

<https://blog.csdn.net/zhanaolu4821/article/details/88894990>