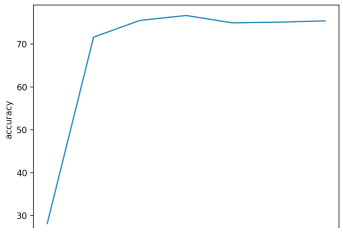
DSP Homework 2-1 Automatic Speech Recognition of Mandarin Digits

R08922A04 林承德

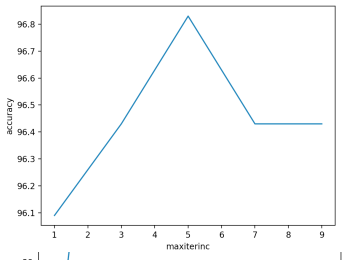
在本次作業中，我們需要使用Kaldi套件，並利用對助教所提供的bash，訓練出一個語音模型，再對其參數進行調整，以期達到最佳的accuracy。

<3-train.sh>

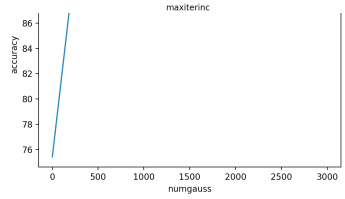
首先我調整了numiters，意思是訓練的iterations數量，從下圖中可以看到在數量較少的時候，performance的上升是急遽的，但當iteration數到6~8左右後就會趨於穩定。在與執行時間權衡後，我最後選用7。



接著，稍微嘗試了一下如果改變maxiterinc，意思是guassian的數量會增加到哪個iteration為止，這邊的實驗顯示，當數值介在約總iterations數量的一半時，可以得到較好的結果。我最後選用5

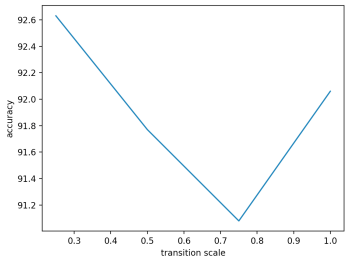


下一個調控的參數是 numgauss 指的是初始的時候，我們的一把gaussian有多少個，由實驗結果可以看到這個數字對於結果有急遽的影響，從原本初始的個位數，一路提升到1000左右的提升幅度最大，接著還會一路提升到將近3000，這是本次作業中非常重要的一項參數。與時間權衡後，我最後選用3000



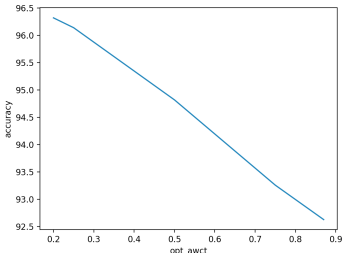
至於totgauss，相較於numgauss的影響則沒有numgauss劇烈，不過這很有可能是因為我的實驗順序是先調升numgauss，若相反過來則有可能導致此項參數與maxiterinc的影響都加劇。不過總理念應該還是圍繞在要用多少個gaussian來逼近我們的observation probability。與時間權衡後，我最後選用4000

最後我還試著調整scale_opts的transition-scales，這邊的幾個參數是gmm在做re-alignment時的各個機率，其中改動transition-scales的結果果如右圖，雖然從圖上看得出來明顯的差別，但其實數值差並不大，有可能是因為某些參數值剛好較適合當前的data。最後選用0.2



<4-test.sh>

在testing時，opt_awct 對 testing的影響是巨大的，由實驗可知，此數值越小，最後的accuracy越高。最後選用0.2



最後是 test_beam，這代表的是最後testing在做beam search時保留的分支數量，可以想像，數值越高可以找到最佳解的機會越大，但同時也會消耗越多的時 間，最後在與時間的權衡下，最後選定50

<topo.proto>

在topo.proto這邊呢，我嘗試改變state的次數分別嚐試了total state = 5，與 total state = 15的情況，但很奇怪的是performance卻下降了，當然有可能是因為資料並不需要那麼多的state，過多的state會導致表現下降，我更傾向的是因為我在調整的時候，所設的initial 參數不佳，或是調整方法不對所導致的。

最後附上最終執行結果：

```
(kaldi) root@33377ede884c:/opt/kaldi/dsp-hw2-1# bash 4-test.sh
Converting acoustic models to HTK format
    viterbi/mono/final.mmf viterbi/mono/tiedlist exist , skipping ...
Generating results for test set with acoustic weight = [ 0.2 ]
    output -> viterbi/mono/test.mlf
    log -> viterbi/mono/log/latgen.test.log
    result -> viterbi/mono/test.rec
    accuracy -> [ 96.55 ] %

Execution time for whole script = 00 hours 00 mins 28 secs
```