语音识别实践

基于HTK的连续语音识别系统

2018/4/8

主要内容

1 HTK简介

2 参数训练

3 语音识别

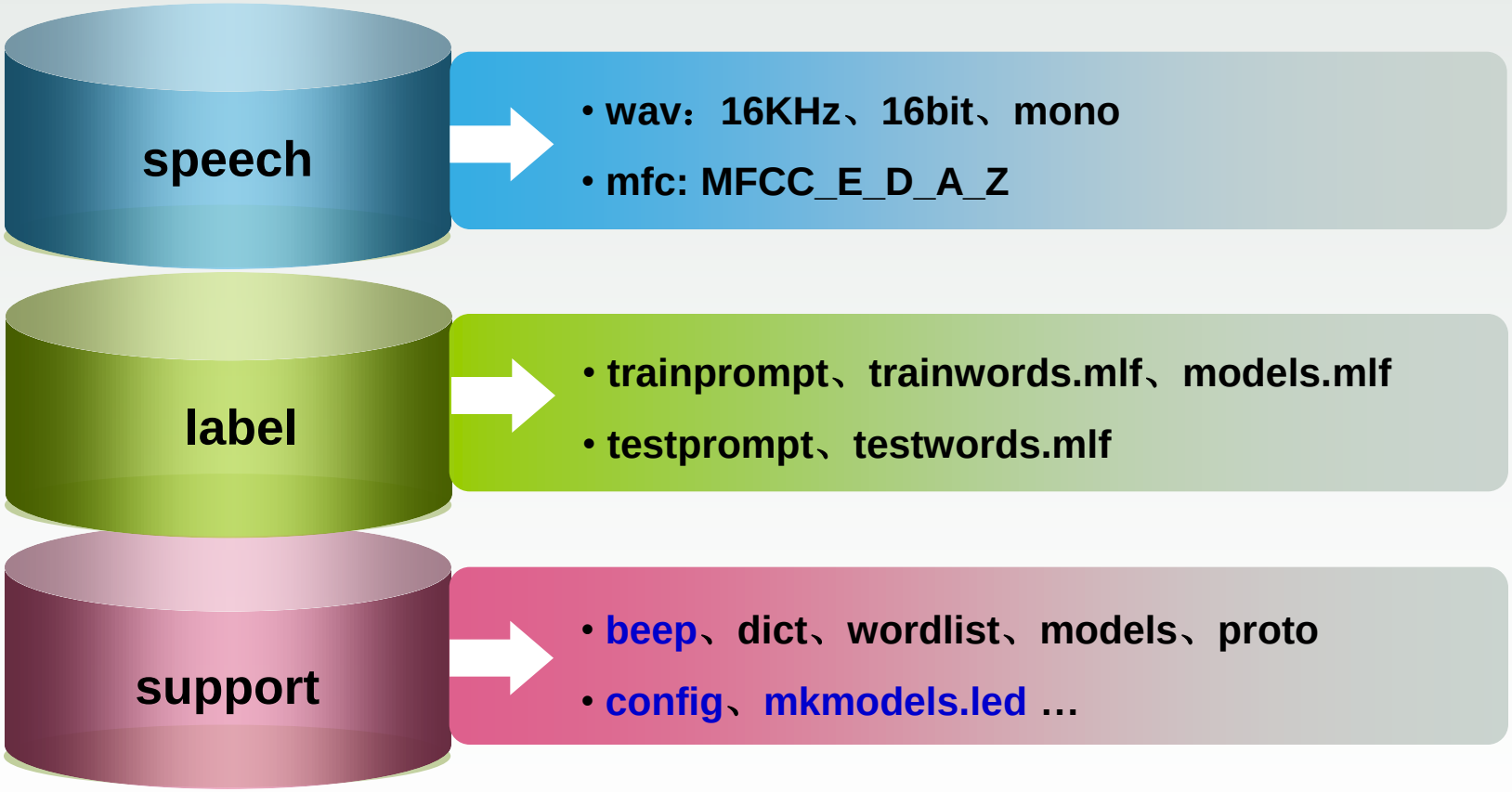
HTK简介

HTK(HMM Toolkit)是一个专门用于建立和处理隐马尔科夫模型的实验工具包。

- 开创：剑桥大学**Steve Young**
- 发展：**Phil Woodland，Entropic Cambridge Research Lab Ltd**
- 壮大：微软合并**Entropic**
- 应用拓展：语音合成(HTS)，字符识别和DNA排序等

模型训练

数据


speech
• wav：16KHz、16bit、mono
• mfc: MFCC_E_D_A_Z
label
• trainprompt、trainwords.mlf、models.mlf
• testprompt、testwords.mlf
support
• beep、dict、wordlist、models、proto
• config、mkmodels.led …

模型

$$\lambda=(N, M, \pi, A, B)$$

声母27： b p m f d t n l g k h j q x zh ch sh r z c s

_a _o _e _y _w _v

韵母36：

单元音韵母9： a、o、e、i （i1、i2）、u、v 、er

复韵母13： ai、ei、ao、ou、ia、ie、ua、uo、ve(üe)、iao、iou、uai、uei

鼻韵母16： an、ian、uan、van(üan)、en、in、uen、vn(ün)、ang、iang、uang、eng、ing、ueng、ong、iong

特殊2： sp sil

模型总数： 65 (+io, 66)

准备数据

- 提取特征

  HCopy -T 1 –C config test.wav test.mfc

  提取语音文件test.wav的特征参数，并保存到文件test.mfc中。

  -T：设置是否跟踪命令

  -C：指定配置文件

  指定目录：

  HCopy -T 1 –C config test.wav data\test.mfc

  HCopy -T 1 –C config test.wav f:\data\test.mfc

  使用脚本：HCopy -T 1 –C config –S codeTr.scp

准备数据

- 发音词典beep
- 词汇列表wordlist
- 由文件beep和wordlist，制作文件models0 dict0

HDMan –w wordlist -n models0 dict0 beeps

准备数据

- 由语句级标注trainprompt，制作词汇级标注trainwords.mlf

  Makewords trainprompt trainwords.mlf

- 由trainwords.mlf制作基元级标注

  HLEd -l * -d dict0 -i models0.mlf mkmodels0.led trainwords.mlf

  -l *：用于在文件名前面加上“*”，为方便分析识别结果。

  -i：将基元级标注写入文件models0.mlf

  mkmodels0.led文件内容：

  EX
  IS sil sil
  DE sp

模型proto

```
~o <VecSize> 39 <MFCC_E_D_A_Z>
~h "proto"
<BeginHMM>
 <NumStates> 5
 <State> 2
      <Mean> 39
          0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
          0.0 0.0 0.0 0.0
      <Variance> 39
          1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
          1.0 1.0 1.0 1.0
 <State> 3
      <Mean> 39
          0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
          0.0 0.0 0.0 0.0
      <Variance> 39
          1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
          1.0 1.0 1.0 1.0
 <State> 4
      <Mean> 39
          0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
          0.0 0.0 0.0 0.0
      <Variance> 39
          1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
          1.0 1.0 1.0 1.0
 <TransP> 5
  0.000e+0    1.000e+0    0.000e+0    0.000e+0    0.000e+0
  0.000e+0    5.000e-1    5.000e-1    0.000e+0    0.000e+0
  0.000e+0    0.000e+0    5.000e-1    5.000e-1    0.000e+0
  0.000e+0    0.000e+0    0.000e+0    5.000e-1    5.000e-1
  0.000e+0    0.000e+0    0.000e+0    0.000e+0    0.000e+0
<EndHMM>
```

训练模型

- 模型proto

- 初始化

HCompV -f 0.01 -m -S train.scp -M hmm0 proto

统计全局数据的均值和方差，并作为所有**HMM**的初始化参数

-f 生成一个方差宏文件，相当于全局方差的**0.01**倍，用于以后的步骤中对方差的评估做一个比较。

-m 使用训练数据的均值作为**HMM**的均值

-S 指定训练阶段使用的脚本，脚本内容指明了参与训练的数据

-M 表示在文件夹**hmm0**中生成两个文件：**proto** 和**vFloors**

proto指明**HMM**结构及初始参数

训练模型

- 制作文件hmmdefs

- 参数重估

HERest -I models0.mlf -t 250.0 150.0 1000.0 -S train.scp -H hmm0\macros -H hmm0\hmmdefs -M hmm1 models

-I：导入标注文件

-t：参数重估修正

-H：导入参数重估前的hmmdefs和macros，即导入重估前的参数

-M：将重估后的HMM写入文件hmmdefs和macros中，并将这个两个文件放置在文件夹hmm1下

训练模型

- sil模型修正

在sil模型中添加从状态2到状态4，以及从状态4到状态2的转换。增加一个sp模型，其参数为sil模型的中间状态。修正sil模型以及增加sp模型为模拟各种类型的噪音，用以增强系统鲁棒性。

(1) 引入global.ded，更新dict文件

HDMan -m -w wordlist -n models1 -l dlog dict1 beep

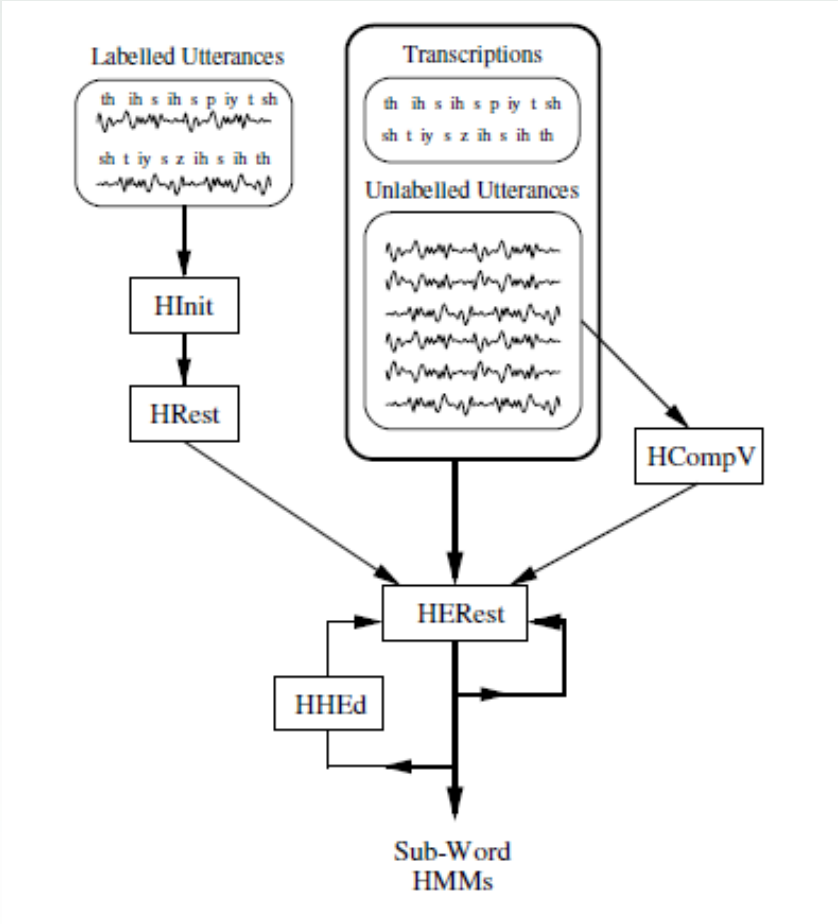
(2) 更新标注文件

HLEd -l * -d dict1 -i models1.mlf mkmodels1.led trainwords.mlf

(3) 修正sil模型

HHEd -H .\Hmms\hmm4\macros -H .\Hmms\hmm4\hmmdefs -M .\Hmms\hmm5 sil.hed models1

HTK训练过程


Labelled Utterances
th ih s ih s p iy t sh
sh t iy s z ih s ih th
Transcriptions
th ih s ih s p iy t sh
sh t iy s z ih s ih th
Unlabelled Utterances
HInit
HRest
HCompV
HERest
HHEd
Sub-Word
HMMs

嵌入式HMM模型训练

- 在单HMM模型参数估计的Baum-Welch算法中，使用**硬边界**，即使用模型对应音段的开始和结束时间。

- 这里的嵌入式HMM模型参数估计Baum-Welch算法，忽略观察序列对应的模型之间的边界，即使用所谓的**软边界**。

优点

- 第一，语音信号的音素切分问题本身就是个不简单的问题，至今还没有一个准确（标准也不好定）的音素自动切分程序，使用嵌入式训练恰好避开了这个麻烦，模型在训练过程中自身会收敛到一个合适的边界。

- 第二，只要给定一个合适的模型初始值，使用软边界训练，得到的模型参数将会更为精确和一致。

✓ **事实上，利用嵌入式训练收敛的HMM参数，可以反过来完成音段的自动切分**。

训练好的HMM

```
~h "a"
<BEGINHMM>
<NUMSTATES> 5
<STATE> 2
<NUMMIXES> 2
<MIXTURE> 1 5.000000e-001
<MEAN> 39
 5.003171e+000 -5.167618e+000 -5.945941e+000 -4.500802e+000 1.315396e+001 3.753041e+000 4.800159e+000 4.108822e-001 3.154825e-001 1.592379e+000 -2.572647e+000 -2.779408e+000
 8.695224e-001 2.091745e-001 1.136440e-001 7.666080e-002 2.994151e-001 4.451593e-001 2.736315e-001 2.585939e-001 2.891368e-001 1.228178e-001 1.686333e-001 2.195840e-001
 1.358487e-001 3.068676e-003 2.624583e-002 1.654327e-001 1.431425e-001 1.957605e-001 -4.895949e-002 4.041954e-002 6.455142e-002 9.616518e-002 1.248790e-001 1.206411e-001
 1.561979e-001 1.394092e-001 -1.035824e-003
<VARIANCE> 39
 1.889638e+001 2.171205e+001 1.842981e+001 4.250087e+001 5.222434e+001 2.535599e+001 3.671383e+001 2.768869e+001 2.824805e+001 2.283212e+001 3.839411e+001 4.312450e+001
 5.400559e-002 6.901817e-001 1.136159e+000 9.172496e-001 1.557579e+000 1.960111e+000 1.402977e+000 1.369985e+000 1.359065e+000 1.274215e+000 1.296601e+000 1.703608e+000
 1.481422e+000 7.685114e-004 1.176687e-001 1.800530e-001 1.604527e-001 2.516406e-001 2.855332e-001 2.682326e-001 2.384717e-001 2.464325e-001 2.257046e-001 2.316903e-001
 2.928817e-001 2.485914e-001 8.877259e-005
<GCONST> 7.806876e+001
<MIXTURE> 2 5.000000e-001
<MEAN> 39
 3.264373e+000 -7.031466e+000 -7.663139e+000 -7.108510e+000 1.026330e+001 1.738851e+000 2.376481e+000 -1.693920e+000 -1.810473e+000 -3.189400e-001 -5.051167e+000 -5.406178e+000
 7.765660e-001 -1.231342e-001 -3.127190e-001 -3.064318e-001 -1.997970e-001 -1.148565e-001 -2.001578e-001 -2.095915e-001 -1.771789e-001 -3.287068e-001 -2.868403e-001 -3.025053e-001
 -3.510061e-001 -8.020138e-003 -1.109657e-001 -4.297952e-003 -1.708369e-002 -4.894704e-003 -2.627007e-001 -1.667452e-001 -1.307828e-001 -1.024027e-001 -6.515448e-002 -7.189579e-002
 -6.027630e-002 -6.002659e-002 -4.804592e-003
<VARIANCE> 39
 1.889638e+001 2.171205e+001 1.842981e+001 4.250087e+001 5.222434e+001 2.535599e+001 3.671383e+001 2.768869e+001 2.824805e+001 2.283212e+001 3.839411e+001 4.312450e+001
 5.400559e-002 6.901817e-001 1.136159e+000 9.172496e-001 1.557579e+000 1.960111e+000 1.402977e+000 1.369985e+000 1.359065e+000 1.274215e+000 1.296601e+000 1.703608e+000
 1.481422e+000 7.685114e-004 1.176687e-001 1.800530e-001 1.604527e-001 2.516406e-001 2.855332e-001 2.682326e-001 2.384717e-001 2.464325e-001 2.257046e-001 2.316903e-001
 2.928817e-001 2.485914e-001 8.877259e-005
<GCONST> 7.806876e+001
<STATE> 3
<NUMMIXES> 2
<MIXTURE> 1 5.000000e-001
<MEAN> 39
 4.429905e+000 -4.362464e+000 -5.357147e+000 -2.197420e+000 1.119998e+001 3.480115e+000 4.481372e+000 1.933128e+000 5.916684e-001 1.561927e+000 -1.661815e+000 -2.268458e+000
 7.876504e-001 1.748359e-001 3.541166e-001 3.782242e-001 5.322772e-001 -5.450131e-002 1.274732e-001 1.413355e-001 4.077413e-001 2.826637e-001 3.226185e-001 2.908365e-001
 2.536668e-001 -1.829224e-003 5.766109e-002 1.580380e-001 1.549675e-001 1.829681e-001 -3.129428e-003 6.326599e-002 7.336428e-002 1.051167e-001 1.225008e-001 1.339168e-001
 1.296396e-001 1.432500e-001 5.229634e-004
<VARIANCE> 39
 3.056053e+001 2.350745e+001 2.092160e+001 5.424197e+001 6.047482e+001 2.931883e+001 3.793505e+001 3.173725e+001 3.311134e+001 2.354764e+001 3.750325e+001 5.060237e+001
 9.995200e-002 7.078421e-001 1.012227e+000 9.884133e-001 1.796542e+000 1.865609e+000 1.593848e+000 1.534554e+000 1.591902e+000 1.377903e+000 1.343125e+000 1.671146e+000
 1.600845e+000 8.876909e-004 1.087514e-001 1.689180e-001 1.695496e-001 3.012073e-001 2.781833e-001 2.797976e-001 2.775881e-001 2.996743e-001 2.492897e-001 2.478624e-001
 3.072366e-001 2.733092e-001 9.918561e-005
<GCONST> 8.207174e+001
```

训练模型

- 增加训练次数

  多次使用HERest命令

- 制作语法文件

  HParse gram wnet

语法设计

设计良好的语法文件有助于提高解码速度和识别精度。例如要识别6位的随机数字串，那么语法文件可以设计为：

```
$digit   = ling | yi | er | san | si | wu | liu | qi | ba | jiu | yao;

( SENT-START ([$digit][$digit][$digit][$digit][$digit][$digit]) SENT-END )
```

其中$用于定义变量，中括号表示固定一个。上述语法文件只定义了一个变量digit，其取值为ling、yi、er、san、si、wu、liu、qi、ba、jiu、yao中的一个。

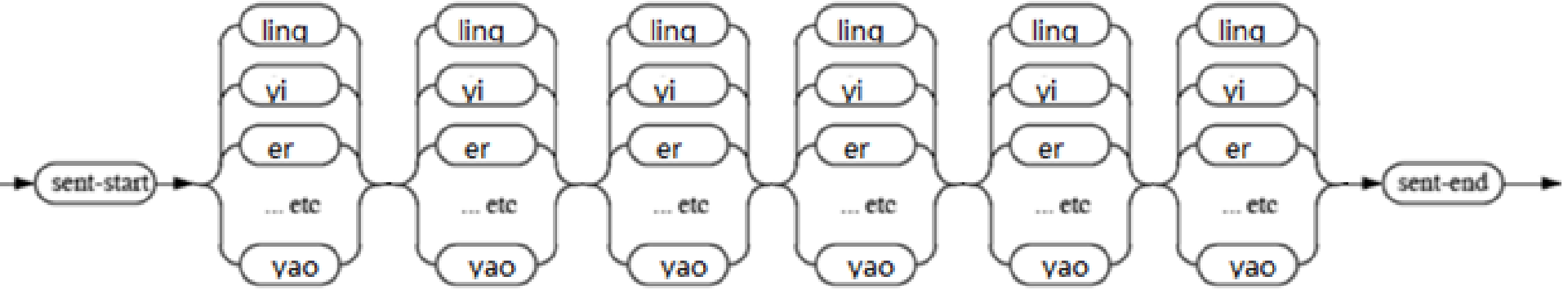
SENT-START定义语句起始，SENT-END定义语句结束。

([$digit][$digit][$digit][$digit][$digit][$digit])定义了语句结构，表明待识别的语句中包含连续的6个digit单元。

语法设计

$digit = ling | yi | er | san | si | wu | liu | qi | ba | jiu | yao;

(SENT-START ([$digit][$digit][$digit][$digit][$digit][$digit]) SENT-END)



语法结构设定后，解码器会把任何内容的语音识别为6位数字串，然后将识别结果和提示文本比对。如果用户朗读的内容并不是提示文本，则无法通过认证。

识别结果（6个数字）

```
#!MLF!#
"*/847306.rec"
0 8900000 SENT-START -4667.023926
8900000 11300000 ba -1926.155640
11300000 14900000 si -2250.643311
14900000 17600000 qi -1983.133911
17600000 21300000 san -2591.906250
21300000 24700000 ling -2233.372559
24700000 26300000 liu -1147.750244
26300000 33100000 SENT-END -3460.816406
.
```

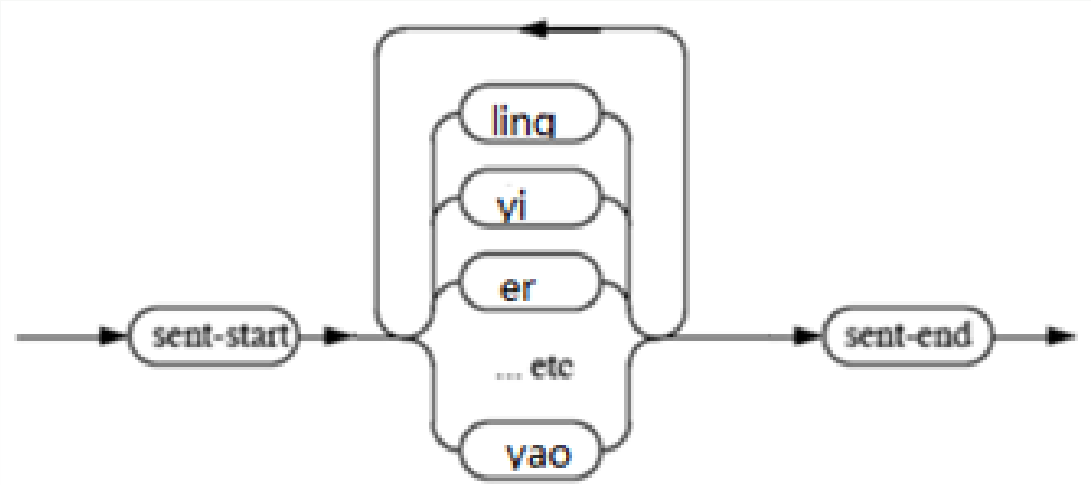
语法设计

将语法设定为固定的长度后，灵活性就降低了，系统无法识别其它长度的数字串。可以对上面的语法稍作更改：

```
$digit  = ling | yi | er | san | si | wu | liu | qi | ba | jiu | yao;
( SENT-START ( <$digit >) SENT-END )
```

尖括号表示可以识别连续的多个重复的digit单元。

这样设计虽然具有较好的灵活性，但识别精度会有所降低。



语法设计

下面给出一个应用于机器人的语法设计。语音控制机器人运动的命令可以有向左转、向右转、向前看、向后看、举右手、举左手等，如果将这些命令一一罗列到语法文件中，语法文件将会复杂而冗长，且不具灵活性。可以按照以下方式设计语法，具体应用可做更复杂的拓展。

定义变量：

方向变量$direction

动作变量$action

动作副词$act

语法设计

$direction = 前 | 后 | 左 | 右 | 上 | 下;

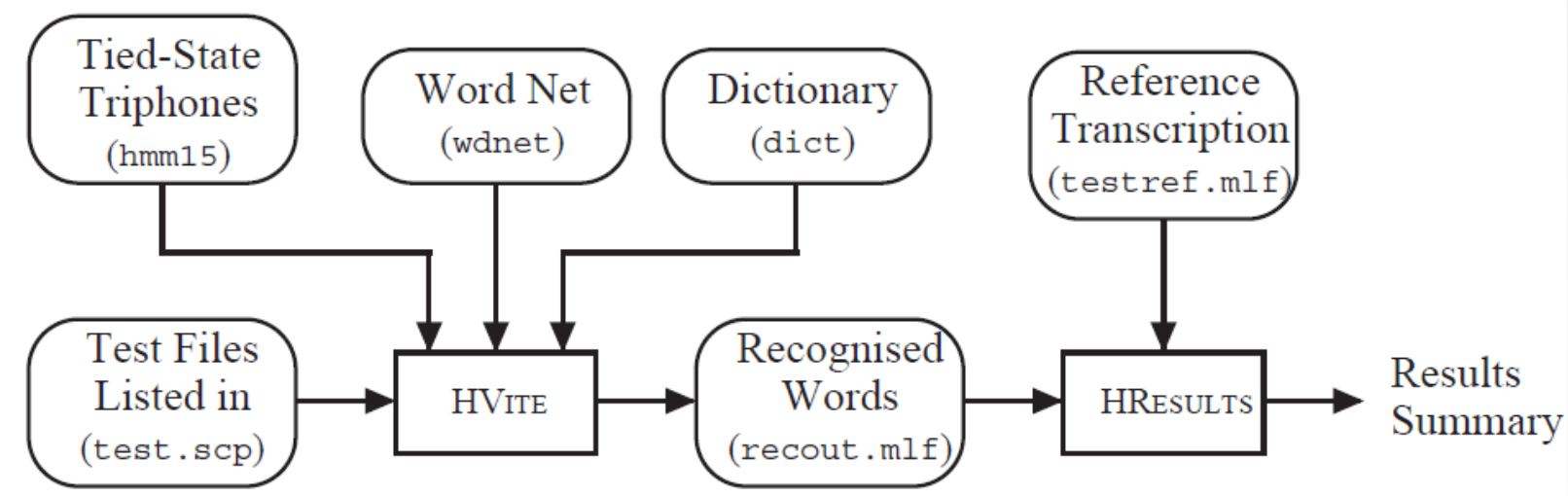
$action = 走 | 看 | 转;

$act = 向;

可以用(SENT-START ($act $direction $action) SENT-END)定义机器人动作向前/后/左/右/上/下看，向前/后/左/右/上/下走,向前/后/左/右转等多个动作。

利用这种方式，可以添加更多的变量值，定义更多更复杂的动作。

识别语音


Tied-State Triphones (hmm15)
Word Net (wdnet)
Dictionary (dict)
Reference Transcription (testref.mlf)
Test Files Listed in (test.scp)
HVITE
Recognised Words (recout.mlf)
HRESULTS
Results Summary

识别语音

- 识别

  HVite -C config_mfc -H hmm10\macros -H hmm10\hmmdefs –S test.scp

  -l * -i recout.mlf -w wnet -p 0.0 -s 5.0 dict models

- 评估分析

  HResults -I testwords.mlf wordlist recout.mlf

```
---------------------- Overall Results -------------------------
SENT: %Correct=50.02 [H=6023, S=6019, N=12042]
WORD: %Corr=90.16, Acc=88.59 [H=108570, D=3621, S=8229, I=1893, N=120420]
================================================================
```

Any more information

please give some questions