

=====

Proyecto: Sistema de reconocimiento de habla en español.
Archivo De Notas

=====

Alumno: Anastópulos Matías
Padrón: 95120

Se deberá armar la siguiente estructura de carpetas:

```
* proyecto
* datos
  * wav
  * train
  * test
  * mfc
    * train
    * test
* etc
* modelos
* rec
* log
* config
* scripts
* lm
NOTAS      (Archivo de texto con todas las explicaciones)
```

Se deberá hacer un link a la base de datos dado a que copiarla generaría un gasto excesivo de memoria. Ejecutar parado en datos:

```
ln -s /dbase/latino40/wav ./wav
```

Agregar al archivo .bashrc el path del HTK:

```
PATH=$PATH:/usr/local/speechapp/htk/bin/
```

Generación de coeficientes Mel-Cepstrum:

Se deberá copiar el archivo go.mfclist. Ejecutar parado en scripts:

```
cp /home/cestien/proyecto/go.mfclist ./
```

Luego ejecutarlo parado en datos:

```
../scripts/go.mfclist genmfc.train genmfc.test
```

Copiarse el archivo de configuracion de HCopy. Ejecutar parado en config:

```
cp /home/cestien/proyecto/config.hcopy ./
```

Ejecutar el HCopy para train y test. Ejecutar parado en datos:

```
HCopy -A -V -T 1 -C ../config/config.hcopy -S genmfc.train > ../log/hcopy.train.log
HCopy -A -V -T 1 -C ../config/config.hcopy -S genmfc.test > ../log/hcopy.test.log
```

Esto generará a partir de los archivos de la base de datos, una base de datos de coeficientes Mel-Cepstrum. Los vectores de coeficientes contendrán los mismos, mas la diferencia entre los mismos y la diferencia de las diferencias. Ademas se incluirá un coeficiente que contenga información de la energía de la señal.

Generación De Master Label Files.

Copiarse en etc los siguientes archivos. Ejecutar parado en etc:

```
cp /home/cestien/proyecto/lexicon ./
cp /home/cestien/proyecto/prompts140.train ./
cp /home/cestien/proyecto/prompts140.test ./
```

Crear el archivo mkphones0.led en etc con el siguiente contenido:

```
EX
IS sil sil
DE sp
```

Esto significa agregar sil al principio y final, modelando silencios largos, y sacar los sp. EX significa reemplazar las palabras por su pronunciacion en el diccionario. Crear el archivo global.ded en etc. Ejecutar el siguiente comando:

```
echo "AS  sp" > global.ded
```

Copiarse el archivo prompts2mlf en scripts. Ejecutar parado en scripts:

```
cp /home/cestien/proyecto/prompts2mlf ./
```

Ejecutar parado en etc:

```
../scripts/prompts2mlf trainmlf prompts140.train
../scripts/prompts2mlf testmlf prompts140.test
```

Esto generará los mlf (Master Label File) de test (testmlf) y train (trainmlf). Ejecutar parado en etc:

```
cat prompts140.train prompts140.test |awk '{for(i=2;i<=NF;i++){print $i}}'|sort|uniq > wlist140
```

Esto generará la lista de palabras (wlistl40). Ejecutar parado en etc: ✓

```
HdMan -m -w wlistl40 -g global.ded -n monophones+sil -l ../log/hdman.log dictl40  lexicon
```

Esto generará los archivos (monophones+sil) y (dictl40). Generará el diccionario dictl40 buscando en el lexicon pronunciaciones de cada palabra en wlistl40. También generará la lista de monophones usados (monophones+sil). También generará un archivo de log con estadísticas e información. ✓

Ejecutar parado en etc:

```
HLEd -l '*' -d dictl40 -i phones0.mlf mkphones0.led trainmlf ✓
```

Que generará el archivo (phones0.mlf), un MLF a nivel monophones.

Creación De Monophones Iniciales.

Parado en config copiarse el archivo config. Ejecutar parado en config:

```
cp /home/cestien/proyecto/config ./
```

Pararse en modelos y copiarse el archio proto. Ejecutar parado en modelos:

```
cp /home/cestien/proyecto/proto ./
```

Este tiene un prototipo de fonema. Pararse en modelos y ejecutar:

```
ls ../datos/mfc/train/*/* > listamfc.txt
```

Que generará el achivo listamfc.txt, la lista con los arhivos que se usan para entrenar. Crear el directorio hmm0 en modelos. Ejecutar parado en modelos:

```
mkdir hmm0
```

Ejecutar parado en modelos:

```
HCompV -C ../config/config -f 0.01 -m -S listamfc.txt -M hmm0 proto ✓
```

Donde proto es el archivo que nos copiamos antes. Esto generará los archivos proto y vfloors en hmm0. Lo que este comando hace es, a partir del archivo de prototipo de fonema, lo reemplaza con uno que tenga las medias y varianzas globales, con el objetivo de tener un punto de partida para la estimación. El archivo de config contiene información respecto al tipo de coeficientes Mel-Cepstrum a utilizar. vfloors contiene información del mínimo para las varianzas. ✓

Primer Entrenamiento.

Copiarse en scripts el archivo go.gen-hmmdefs. Ejecutar parado en scripts:

```
cp /home/cestien/proyecto/go.gen-hmmdefs ./
```

Editar el archivo monophones+sil borrando al linea de sp. Ejecutar parado en etc:

```
cp monophones+sil monophones+sil2
```

Que generará monophones+sil2. Ejecutar parado en etc:

```
vim monophones+sil2
```

Y borrar la linea de sp. *y agregar sil* Pararse en modelos y ejecutar:

```
../scripts/go.gen-hmmdefs ../etc/monophones+sil2 hmm0/proto > hmm0/hmmdefs ✓
```

Que generará el archivo hmmdefs en hmm0. Los siguiente es copiarse el archivo(go.gen-macros). Ejecutar parado en scripts:

```
cp /home/cestien/proyecto/go.gen-macros ./
```

Ejecutar en modelos/hmm0:

```
../scripts/go.gen-macros vfloors proto > macros
```

Que generará el archivo macros. Crear en modelos el directorio hmm1:

```
mkdir hmm1
```

Ejecutar el primer entrenamiento parado en modelos: (Agregar el sil en monophones+sil2 con el vim). *(antes, si no go.genhmm no lo inicieli ya)*

```
HERest -C ../config/config -I ../etc/phones0.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm0/macros -H hmm0/hmmdefs -M hmm1 ../etc/monophones+sil2 ✓
```

Crear el directorio hmm2 y hacer el segundo entrenamiento. Ejecutar parado en modelos:

```
mkdir hmm2
```

```
HERest -C ../config/config -I ../etc/phones0.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm1/macros -H hmm1/hmmdefs -M hmm2 ../etc/monophones+sil2 ✓
```

Ahora hemos realizado con HERest los dos primeros entrenamientos de fonemas, los modelos resultantes quedaran guardados en los sucesivos archivos hmmX.

Modelo De Pausa Corta:

Lo siguiente es agregar el sp al modelo para poder modelar silencios cortos en habla continúa. Utilizaremos en lugar de un fonema de 5 estados para modelarlo, un de 3 en su lugar. Copiarse los archivos de hmm2 en un hmm3, ejecutar parado en modelos:

```
mkdir hmm3
cd hmm3
cp ../hmm2/hmmdefs ./
cp ../hmm2/macros ./
```

Editar con el vim el hmmdefs para incluir el modelo de sp. Se deberá copiar el modelo del sil y pegarlo al final tal cual esta. Se deberá cambiar el nombre del nuevo de sil a sp. Se deberá eliminar los estados 2 y 4 y al 3 llamarlo 2. Cambiar todos los lugares donde diga 5 (Estados) a 3. Editar la matriz de transición eliminando las dos ultimas columnas y las dos anteultimas filas (De modo que quede una matriz de transición normal, pero con una fila de ceros al final). Luego parado en etc generar un nuevo archivo de monophones que tenga sp al final. ✓

```
cp monophones+sil2 monophones+sil3
```

↳silencio que pueden ocurrir o no entre palabras

```
vim monophones+sil3 ✓
```

Y agregar el sp al final. Luego crear el archivo (sil.hed) en modelos con el siguiente contenido:

```
AT 2 4 0.2 {sil.transP}  
AT 4 2 0.2 {sil.transP}  
AT 1 3 0.3 {sp.transP}  
TI silst {sil.state[3],sp.state[2]}
```

→ *ya agrega otras transiciones entre estados del silencio y de sp.*

Esto hará que el estado central de sil y el estado central de sp se estimen juntos. Crear el directorio hmm4 en modelos:

```
mkdir hmm4
```

Ejecutar parado en modelos:

```
HHed -H hmm3/macros -H hmm3/hmmdefs -M hmm4 sil.hed ../etc/monophones+sil3 ✓
```

Que generará archivos en hmm4 (hmmdefs y macros), con las instrucciones de sil.hed. Pararse en etc y crear un nuevo archivo de mkphones1.led:

```
cp mkphones0.led mkphones1.led  
vim mkphones1.led
```

Sacarle la linea que dice "DE sp" para que no borre el sp. Ejecutar parado en etc: ✓

```
HLEd -l '*' -d dictl40 -i phones1.mlf mkphones1.led trainmlf
```

Que generará phones1.mlf, el nuevo MLF a nivel fonemas. *con sp* Pararse en modelos y generar el directorio hmm5:

```
mkdir hmm5
```

Ejecutar parado en modelos otro entrenamiento (Ahora con phones1.mlf): ✓

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm4/macros -H hmm4/hmmdefs -M hmm5 ../etc/monophones+sil3
```

Crear el directorio hmm6. Ejecutar parado en modelos:

```
mkdir hmm6
```

Y realizar otro entrenamiento: ✓

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm5/macros -H hmm5/hmmdefs -M hmm6 ../etc/monophones+sil3
```

Modelo de lenguaje.

Agregar ngram-count al PATH en .bashrc, esta en el directorio /usr/local/speechapp/srilm/bin/i686-m64/. Ejecutar parado en etc:

```
awk '{for(n=2;n<=NF;n++){printf "%s ",$n;};printf "\n";}' promptsl40.train > train.txt ✓
```

Que generará la lista de oraciones, sin la etiqueta del archivo a la que corresponde cada uno. Ejecutar parado en etc:

```
cat promptsl40.test |awk '{for(i=2;i<=NF;i++){print $i}}'|sort|uniq > vocab ✓
```

Para generar el archivo de volcabulario. Ejecutar parado en etc: ✓

```
ngram-count -order 2 -text train.txt -lm lml40 -ukndiscount2 -vocab vocab
```

Esto genera el archivo lml40 que contiene cada par de palabras junto con sus probabilidades.

```
Agregar a vocab <s> y </s>.  
Agregar al dictl40:  
<s> [] sil  
</s> [] sil ✓
```

Lo siguiente a realizar será convertir esta estadística al formato del HTK. Para pasar a formato del HTK los bigramas ejecutar desde etc:

```
HBuild -s '<s>' '</s>' -n lml40 vocab wdnnet ✓
```

Que generará el archivo wdnnet con la estadística. *red de palabras con probabilidades de transición*

Un primer reconocimiento.

Generar el archivo (test.scf), ejecutar en modelos:

```
ls ../datos/mfc/test/*/* > test.scf
```

Que genera la lista de archivos de test. Hacer el Viterbi, ejecutar desde la carpeta rec:

lg →

```
HVite -C ../config/config -H ../modelos/hmm6/macros -H ../modelos/hmm6/hmmdefs -S ../modelos/test.scf -l '*' -i recout.mlf -w ../etc/wdnnet -p 0.0 -s 5.0 \
```

Indicando los fonemas estimados en el hmm6, que es el último realizado. El resultado será un MLF recout.mlf con lo que intentó reconocer.

Ver los resultados ejecutando desde rec el siguiente comando:

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout.mlf > resultados1.txt
```

Debería ser de al rededor del 50% ya que se necesitan agregar mas gaussianas.

Agregar gaussianas:

Lo que sigue serán una serie de iteraciones en las cuales se mejorará la efectividad del reconocimiento agregando mas combinaciones de gaussianas a cada estado.

Crear en modelos el directorio hmm7. Ejecutar en modelos:

```
mkdir hmm7
```

Crear el archivo split1.led en modelos usando el vim, con la siguiente linea:

```
MU 2 {*.state[2-4].mix}
```

Esto significa que parta cada gaussiana en dos. Ejecutar el editor, en modelos ejecutar:

```
HHed -H hmm6/macros -H hmm6/hmmdefs -M hmm7 split1.led ../etc/monophones+sil3
```

Arroja el siguiente warning:

```
WARNING [-2637] HeaviestMix: mix 1 in sil has v.small gConst [60.084660] in HHed
```

Hacer el reentrenamiento 2 veces mas. Ejecutar en modelos:

```
mkdir hmm8
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm7/macros -H hmm7/hmmdefs -M hmm8 ../etc/monophones+sil3
```

```
mkdir hmm9
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm8/macros -H hmm8/hmmdefs -M hmm9 ../etc/monophones+sil3
```

Volver a ejecutar HVite. Ejecuar en rec:

```
HVite -C ../config/config -H ../modelos/hmm9/macros -H ../modelos/hmm9/hmmdefs -S ../modelos/test.scp -l '*' -i recout2.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

Volver a ver los resultados. Ejecutar en rec:

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout2.mlf > resultados2.txt
```

Repetir el anterior hasta que no mejore mas la efectividad. Lo que sigue es la secuencia de iteraciones realizadas hasta que la efectividad no mejore mas.

==== ITERACION 2 =====

En modelos:

```
vim split2.led
```

```
MU 4 {*.state[2-4].mix}
```

```
mkdir hmm10
```

```
HHed -H hmm9/macros -H hmm9/hmmdefs -M hmm10 split2.led ../etc/monophones+sil3
```

```
mkdir hmm11
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm10/macros -H hmm10/hmmdefs -M hmm11 ../etc/monophones+sil3
```

```
mkdir hmm12
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm11/macros -H hmm11/hmmdefs -M hmm12 ../etc/monophones+sil3
```

En rec:

```
HVite -C ../config/config -H ../modelos/hmm12/macros -H ../modelos/hmm12/hmmdefs -S ../modelos/test.scp -l '*' -i recout3.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout3.mlf > resultados3.txt
```

==== ITERACION 3 =====

En modelos:

```
vim split3.led
```

```
MU 8 {*.state[2-4].mix}
```

```
mkdir hmm13
```

```
HHed -H hmm12/macros -H hmm12/hmmdefs -M hmm13 split3.led ../etc/monophones+sil3
```

```
mkdir hmm14
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm13/macros -H hmm13/hmmdefs -M hmm14 ../etc/monophones+sil3
```

```
mkdir hmm15
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm14/macros -H hmm14/hmmdefs -M hmm15 ../etc/monophones+sil3
```

En rec:

```
HVite -C ../config/config -H ../modelos/hmm15/macros -H ../modelos/hmm15/hmmdefs -S ../modelos/test.scp -l '*' -i recout4.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout4.mlf > resultados4.txt
```

==== ITERACION 4 =====

En modelos:

```
vim split4.led
```

```
MU 16 {*.state[2-4].mix}
```

```
mkdir hmm16
```

```
HHed -H hmm15/macros -H hmm15/hmmdefs -M hmm16 split4.led ../etc/monophones+sil3
```

```
mkdir hmm17
```

→ en realidad significa que agregue gaussianas (copiando las que hay, un poquito cambiadas) hasta llegar a ?

2g →

4g →

8g →

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm16/macros -H hmm16/hmmdefs -M hmm17 ../etc/monophones+sil3
mkdir hmm18
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm17/macros -H hmm17/hmmdefs -M hmm18 ../etc/monophones+sil3
```

En rec:

```
HVite -C ../config/config -H ../modelos/hmm18/macros -H ../modelos/hmm18/hmmdefs -S ../modelos/test.scp -l '*' -i recout5.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout5.mlf > resultados5.txt
```

Hasta aca el resultado es:

```
----- Overall Results -----
SENT: %Correct=35.66 [H=352, S=635, N=987]
WORD: %Corr=80.54, Acc=75.31 [H=6371, D=268, S=1271, I=414, N=7910]
=====
```

==== ITERACION 5 =====

En modelos:

```
vim split5.led
```

```
MU 32 {*.state[2-4].mix}
```

```
mkdir hmm19
```

```
HHed -H hmm18/macros -H hmm18/hmmdefs -M hmm19 split5.led ../etc/monophones+sil3
```

```
mkdir hmm20
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm19/macros -H hmm19/hmmdefs -M hmm20 ../etc/monophones+sil3
```

```
mkdir hmm21
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm20/macros -H hmm20/hmmdefs -M hmm21 ../etc/monophones+sil3
```

En rec:

```
HVite -C ../config/config -H ../modelos/hmm21/macros -H ../modelos/hmm21/hmmdefs -S ../modelos/test.scp -l '*' -i recout6.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout6.mlf > resultados6.txt
```

Hasta aca el resultado es:

```
----- Overall Results -----
SENT: %Correct=41.64 [H=411, S=576, N=987]
WORD: %Corr=83.15, Acc=79.30 [H=6577, D=223, S=1110, I=304, N=7910]
=====
```

==== ITERACION 6 =====

En modelos:

```
vim split6.led
```

```
MU 64 {*.state[2-4].mix}
```

```
mkdir hmm22
```

```
HHed -H hmm21/macros -H hmm21/hmmdefs -M hmm22 split6.led ../etc/monophones+sil3
```

```
mkdir hmm23
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm22/macros -H hmm22/hmmdefs -M hmm23 ../etc/monophones+sil3
```

```
mkdir hmm24
```

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm23/macros -H hmm23/hmmdefs -M hmm24 ../etc/monophones+sil3
```

En rec:

```
HVite -C ../config/config -H ../modelos/hmm24/macros -H ../modelos/hmm24/hmmdefs -S ../modelos/test.scp -l '*' -i recout7.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3
```

```
HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout7.mlf > resultados7.txt
```

Hasta aca el resultado es:

```
----- Overall Results -----
SENT: %Correct=45.39 [H=448, S=539, N=987]
WORD: %Corr=84.69, Acc=81.21 [H=6699, D=207, S=1004, I=275, N=7910]
=====
```

==== ITERACION 7 =====

En modelos:

```
vim split7.led
```

```
MU 128 {*.state[2-4].mix}
```

```
mkdir hmm25
```

```
HHed -H hmm24/macros -H hmm24/hmmdefs -M hmm25 split7.led ../etc/monophones+sil3
```

```
mkdir hmm26
```

1288
→

```
HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm25/macros -H hmm25/hmmdefs -M hmm26 ../etc/monophones+sil3
mkdir hmm27

HERest -C ../config/config -I ../etc/phones1.mlf -t 250.0 150.0 1000.0 -S listamfc.txt -H hmm26/macros -H hmm26/hmmdefs -M hmm27 ../etc/monophones+sil3

En rec:

HVite -C ../config/config -H ../modelos/hmm27/macros -H ../modelos/hmm27/hmmdefs -S ../modelos/test.scp -l '*' -i recout8.mlf -w ../etc/wdnet -p 0.0 -s 5.0 \
../etc/dictl40 ../etc/monophones+sil3

HResults -t -f -I ../etc/testmlf ../etc/wlistl40 recout8.mlf > resultados8.txt

Hasta aca el resultado es:

----- Overall Results -----
SENT: %Correct=46.10 [H=455, S=532, N=987]
WORD: %Corr=85.44, Acc=81.64 [H=6758, D=202, S=950, I=300, N=7910]
=====
```



Perfecto!!