Procesamiento Del Habla

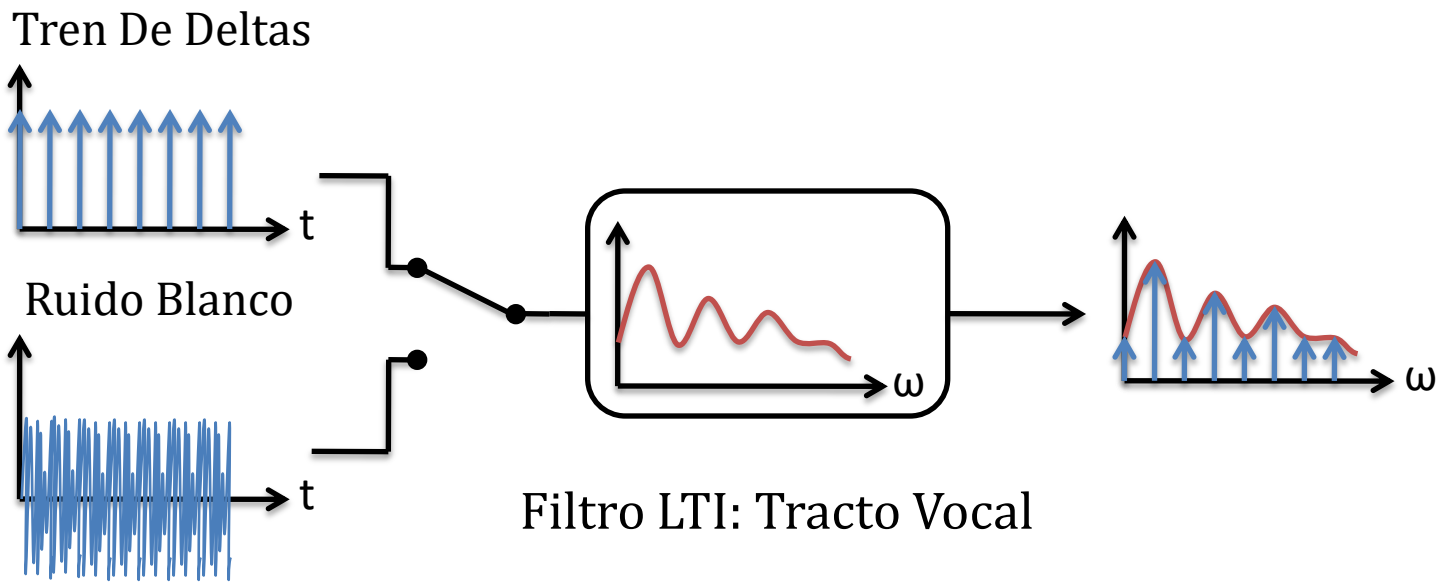
Implementación De Un Reconocedor De Voz

Alumno: Anastópulos Matías

Padrón: 95120

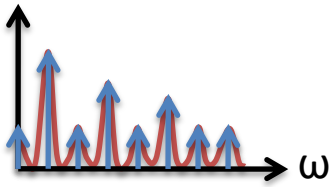
Cuatrimestre De Cursada: 1ero Del 2018

Modelo De Producción Del Habla

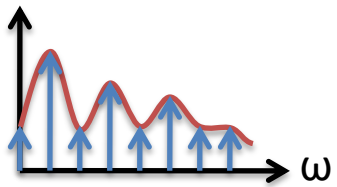


- La señal de habla no es estacionaria.

Pero se supondrá estacionaria dentro de determinadas ventanas de tiempo.



Ventanas Grandes -> Frecuencia Glótica



Ventanas Pequeñas -> Envolvente

- La señal de habla tiene redundancia.

Se deberá aplicar alguna técnica para obtener la información fundamental de la misma. (LPC o Cepstrum)

Linear Predictive Coding

Vamos a tratar de predecir cada muestra de la señal en función de las anteriores.

Estimador Lineal: $s[n] = \sum_{i=1}^{M} b_i s[n-i]$

Buscamos minimizar el ECM => Filtro De Wiener

$$\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} = \begin{bmatrix} \rho(0) & \rho(1) & \dots & \rho(M-1) \\ \rho(1) & \rho(0) & \dots & \rho(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ \rho(M-1) & (M-2) & \dots & \rho(0) \end{bmatrix}^{-1} \begin{bmatrix} \rho(1) \\ \rho(2) \\ \vdots \\ \rho(M) \end{bmatrix}$$

Se deberá utilizar algún estimador de la autocorrelación ρ(i).

Aplicación A La Compresión De Datos Para Las Comunicaciones

La técnica consiste en enviar en lugar de la señal original:

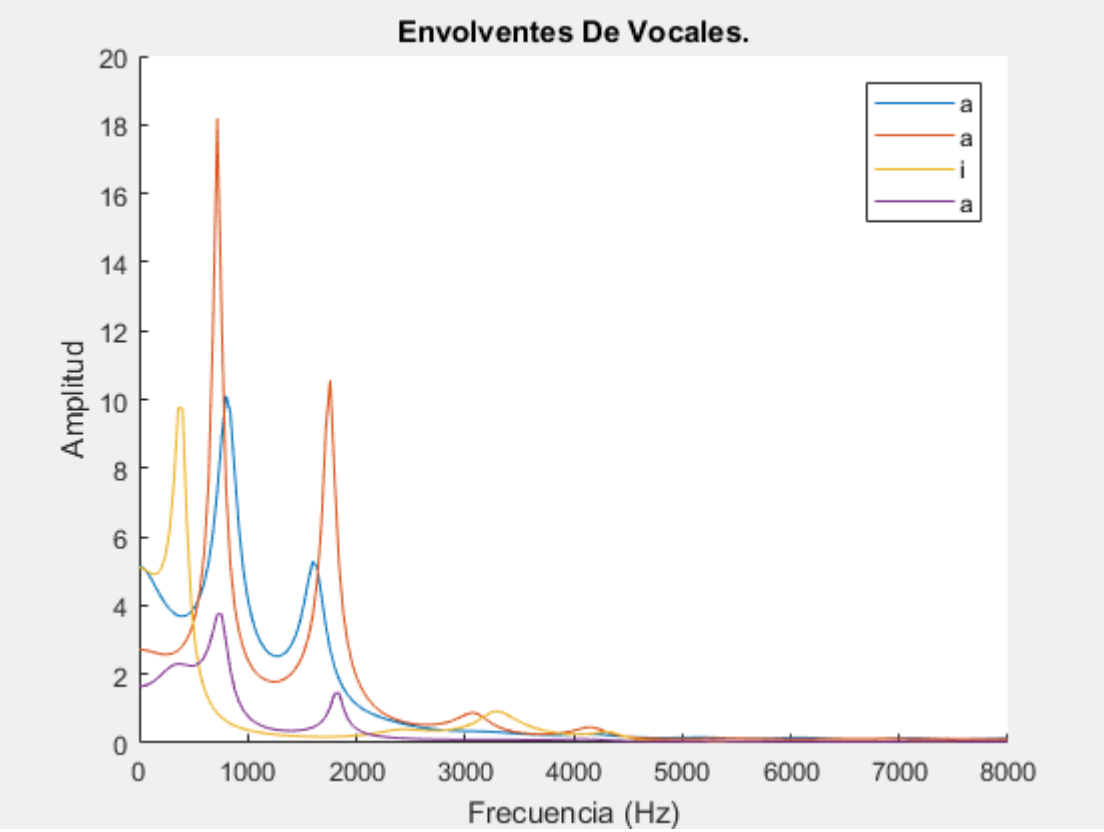
- Los coeficientes LPC.
- Las primeras muestras.
- La señal de error.

Todo para cada ventana. De esta forma el receptor podrá reconstruir la señal original.

$$Coeficientes\ LPC \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} \qquad Señal\ de\ error\ e[n] = s[n] - \widehat{s[n]}$$

Los coeficientes LPC dan información sobre el tracto vocal.

$$H(z) = \frac{G}{1 - \sum_{i=1}^{M} b_i z^{-i}} \qquad G^2 = \rho(0) - \boldsymbol{b}^{\boldsymbol{T}} \boldsymbol{\rho}$$



Los picos o frecuencias de resonancia las llamaremos **formantes**.

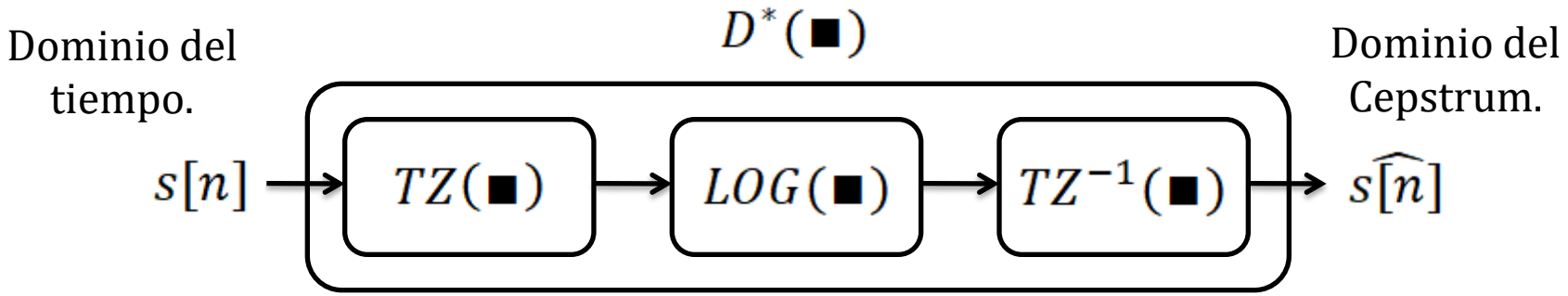
En principio los coeficientes LPC servirían para identificar o clasificar el fonema que esta siendo producido.

Cepstrum

La señal de habla tiene información sobre la frecuencia glótica y sobre el filtro del tracto vocal. Sólo la parte del tracto vocal es relevante para el reconocimiento.
Cepstrum es una técnica para poder separar la parte relevante para el reconocimiento de voz de la que no lo es.

Consiste en llevar la señal a un dominio donde ambas señales sean fáciles de separar.

Consideremos el siguiente operador:



$$s[n] = x[n] * h[n]$$

$x[n]$: *Pulso Glótico*
$h[n]$: *Tracto Vocal*

$$S(z) = X(z)H(z)$$

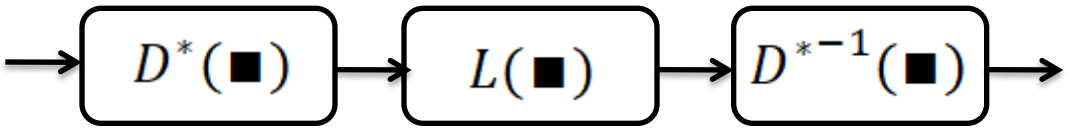
$$\log[S(z)] = \log[X(z)] + \log[H(z)]$$

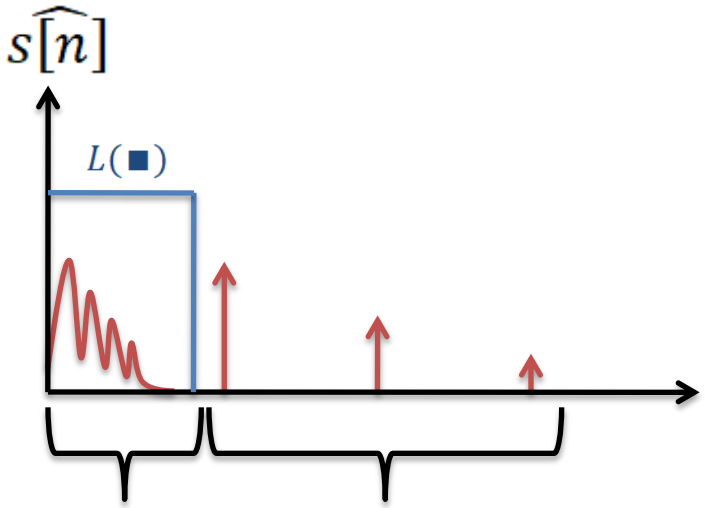
$$\mathrm{TZ}^{-1}\{\log[S(z)]\} = \mathrm{TZ}^{-1}\{\log[X(z)]\} + \mathrm{TZ}^{-1}\{\log[H(z)]\}$$

$$\widehat{s[n]} = \widehat{x[n]} + \widehat{h[n]}$$

En este dominio del Cepstrum, donde aparecen sumadas, son mas fáciles de separar.

Filtrado En El Dominio Del Cepstrum





Información Sobre El Tracto Vocal

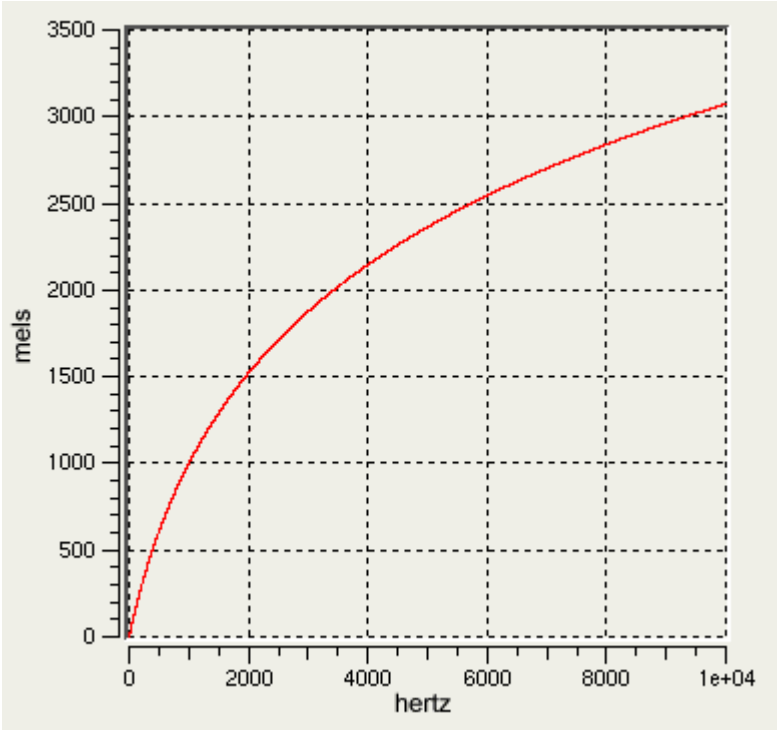
Información Sobre La Frecuencia Glótica

Consideraciones

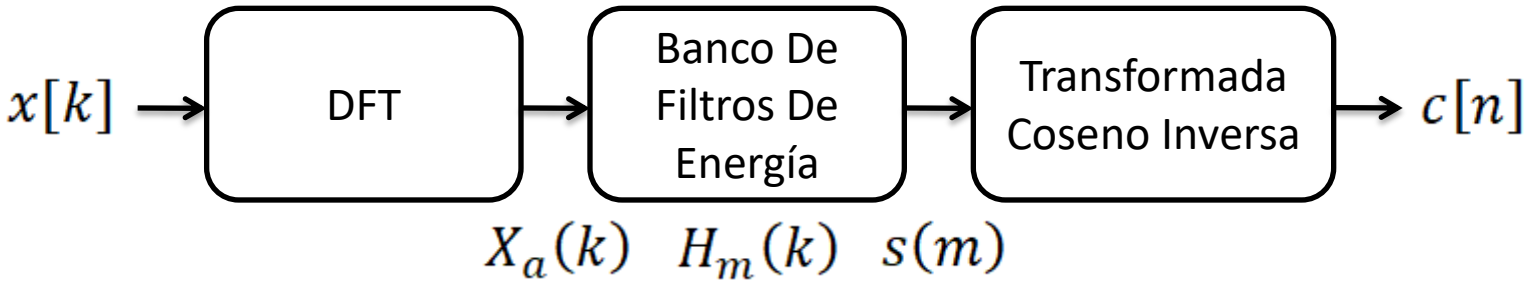
En la práctica utilizamos DFT en lugar de transformada Z.

Los coeficientes Cepstrum son superiores a los LPC para reconocimiento ya que permiten incorporar ciertas características de la audición:

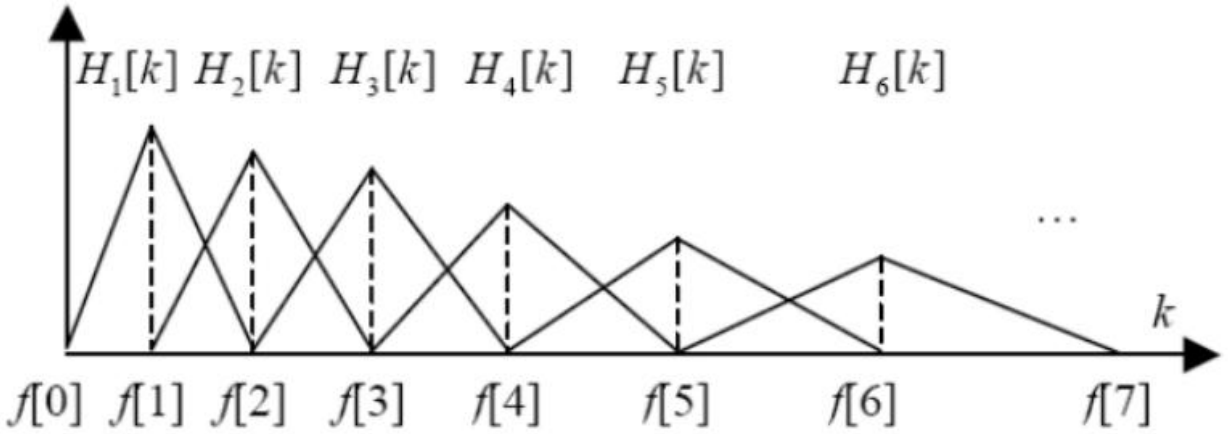
- Escala Mel: Es una relación entre la frecuencia producida y la percibida. Puede observarse que no es lineal.
- Bandas Críticas: Es oído tiene una cierta capacidad de percibir dos frecuencias diferentes, pero por debajo de esa capacidad las percibe como iguales. El ancho de banda crítico varia con la frecuencia.



Coeficientes Mel-Cepstrum



$$s(m) = \log\left[\sum_{k=0}^{N-1} |X_a(k)|^2 H_m(k)\right] \qquad c[n] = \sum_{m=0}^{M-1} s[m]\cos\left[\frac{\pi n(m+\frac{1}{2})}{M}\right]$$



Clasificación

Sea $x \in \mathbb{R}^D$ se lo desea clasificar en alguna categoría C_k .

Criterio De Decisión:

$$\hat{k} = \underset{\text{k}}{\arg\max}\, P(C_k|x) = \underset{\text{k}}{\arg\max}\, P(x|C_k)P(C_k)$$

Donde supondremos:

$$P(x|C_k) \approx N(\mu_k, \Sigma_k)$$

Aprendizaje Supervisado

Conjunto de entrenamiento:

$$D = \{(x^{(1)}, z^{(1)}), (x^{(2)}, z^{(2)}) \dots (x^{(m)}, z^{(m)})\}$$

$x: muestra.$
$z: clase.$

Aplicamos máxima verosimilitud para obtener los parámetros de las gaussianas.

$$\mu_k = \frac{1}{m_k} \sum_{n=1}^{m_k} x^{(n)} \qquad \Sigma_k = \frac{1}{m_k} \sum_{n=1}^{m_k} (x^{(n)} - \mu_k)(x^{(n)} - \mu_k)^T$$

Donde para cada parámetro se usan las muestras que pertenecen a esa clase en particular.

Aprendizaje No Supervisado - EM

Conjunto de entrenamiento: $D = \{x^{(1)}, x^{(2)} \dots x^{(m)}\}$

Aparece la función de responsabilidad:

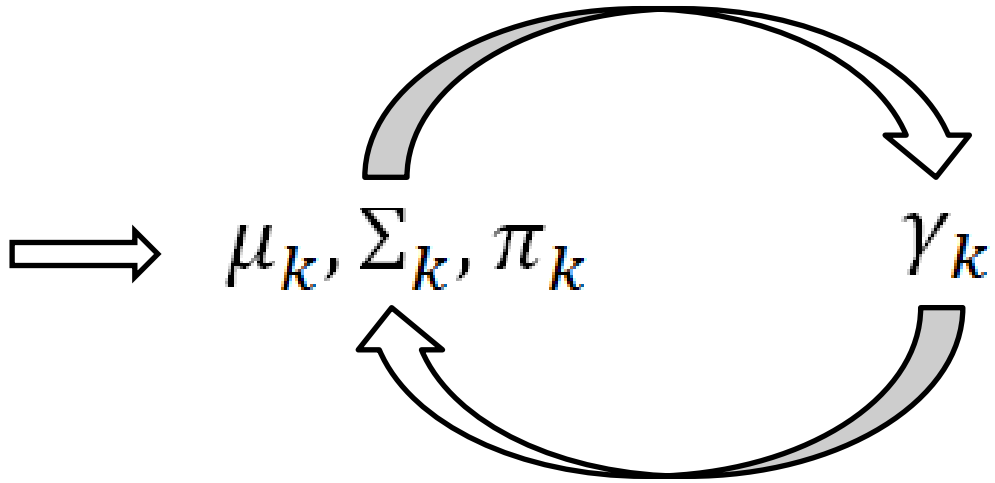
$$\gamma_k(x) = P(z = k|x) = \frac{\pi_k N(x|\mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_k N(x|\mu_k, \Sigma_k)} \qquad \pi_k = P(z = k)$$

Maximizando el LL se llega a los siguientes estimadores:

$$\mu_k = \frac{\sum_{i=1}^{m} \gamma_k(x^{(i)}) x^{(i)}}{\sum_{i=1}^{m} \gamma_k(x^{(i)})} \qquad \pi_k = \frac{1}{m} \sum_{i=1}^{m} \gamma_k(x^{(i)})$$

$$\Sigma_k = \frac{\sum_{i=1}^{m} \gamma_k(x^{(i)})(x^{(i)} - \mu_k)(x^{(i)} - \mu_k)^T}{\sum_{i=1}^{m} \gamma_k(x^{(i)})}$$

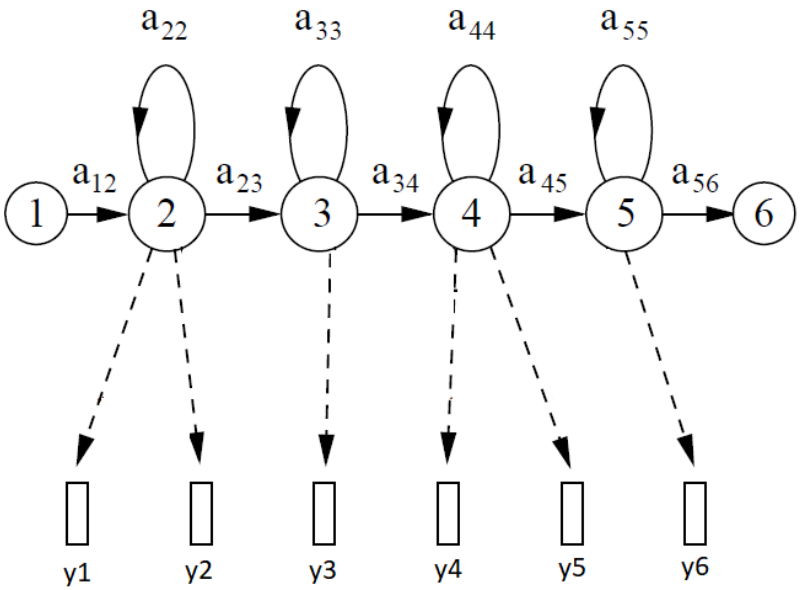
Es un proceso iterativo.



Se repite hasta que el LL no se incremente mas.

$$LL = logP(\{x^{(1)}, x^{(2)} \dots x^{(m)}\}) = \sum_{i=1}^{m} log \left\{ \sum_{k=1}^{K} \pi_k N(x^{(i)} | \mu_k, \Sigma_k) \right\}$$

Modelos Ocultos De Markov



• Es un autómata.

• La transición entre estados estocástica, y la probabilidad de cambiar a un estado siguiente depende únicamente del estado actual.

• En cada estado el autómata realiza la emisión de un valor o característica, con una distribución que depende únicamente del estado actual.

Hipótesis

$$P(q_t|q_{t-1}, \cdots, q_1) = P(q_t|q_{t-1})$$

$$P(y_t|q_t, \cdots, q_1, y_{t-1}, \cdots, y_1) = P(y_t|q_t)$$

Definición

HMM: $M = (a_{i,j}, \pi_i, b_j)$

Matriz De Transición: $a_{i,j} = P(q_t = j|q_{t-1} = i)$

Estado Inicial: $\pi_i = P(q_1 = i)$

Distribución De Las Emisiones: $b_j(y_t) = P(y_t|q_t = j)$

Notación: $\mathbb{Y} = \{y_1, y_2, \cdots, y_T\}$ Secuencia de emisiones.

$\mathbb{Q} = \{q_1, q_2, \cdots, q_T\}$ Secuencia de estados.

Preguntas

1) Dadas las emisiones, poder calcular:

$$\mathbb{Y} = \{y_1, y_2, \dots, y_T\} \longrightarrow P(\mathbb{Y}|M)$$

(Reconocimiento De Habla Aislada)

2) Dadas las emisiones, poder estimar la secuencia de estados mas probable (Viterbi):

$$\mathbb{Y} = \{y_1, y_2, \dots, y_T\}$$

$$\widehat{\mathbb{Q}} = \{\widehat{q_1}, \widehat{q_2}, \dots, \widehat{q_T}\} = \underset{\forall \mathbb{Q}}{\arg max}\, P(\mathbb{Q}|\mathbb{Y})$$

(Reconocimiento De Habla Conectada Y Continua)

3) Estimar los parámetros de un modelo (EM):

$$M = (a_{i,j}, \pi_i, b_j)$$

Recursión Forward Backward

$$P(\mathbb{Y}) = \sum_{\forall \mathbb{Q}} P(y_1, \dots, y_T, q_1, \dots, q_T) \longrightarrow$$ Intratable

Se define:

$$\alpha_t(j) = P(y_1, \dots, y_t, q_t = j)$$

$$\beta_t(i) = P(y_{t+1}, \dots, y_T, q_t = i)$$

Que se pueden calcular con el siguiente algoritmo:

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}\, a_{i,j} b_j(y_t) \qquad j = 1, \dots, N(Cantidad\ de\ estados)$$
$$t = 2, \dots, T(Tiempos)$$

$$\alpha_1(i) = \pi_i b_i(y_1) \longrightarrow$$ Se calcula avanzando en el tiempo.

$$\beta_t(i) = \sum_{j=1}^{N} a_{i,j}\, b_j(y_{t+1}) \beta_{t+1}(j) \qquad i = 1, \dots, N(Cantidad\ de\ estados)$$
$$t = 1, \dots, T-1(Tiempos)$$

$$\beta_T(i) = \frac{1}{N} \longrightarrow$$ Se calcula retrocediendo en el tiempo.

Recursión Forward Backward

$$P(\mathbb{Y}|M) = \sum_{i=1}^{N} \alpha_T(i) \longrightarrow$$ Primer pregunta resuelta.

Será útil definir también:

$$\gamma_t(i) = P(q_t = i|\mathbb{Y}) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}$$

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j|\mathbb{Y}) = \frac{\alpha_t(i)a_{i,j}b_j(y_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}$$

Viterbi

Es un algoritmo para obtener el camino óptimo:

$$\widehat{\mathbb{Q}} = \{\widehat{q_1}, \widehat{q_2}, \dots, \widehat{q_T}\} = \underset{\forall \mathbb{Q}}{\arg max}\, P(\mathbb{Q}|\mathbb{Y})$$

Se define:

$$\varphi_t(j) = \underset{\forall \mathbb{Q}_{t-1}}{max} P(y_1, \dots, y_t, q_1, \dots, q_{t-1}, q_t = j)$$

Algoritmo:

Inicialización:

$$\varphi_1(i) = \pi_i b_1(y_1) \qquad i = 1, \dots, N$$

$$\psi_1(i) = 0$$

Viterbi

Recursión:

$$\varphi_t(j) = \max_{1 \le i \le N} [a_{i,j}\varphi_{t-1}(i)]b_j(y_t) \qquad t = 2, \dots, T$$

$$\psi_t(i) = \arg\max_{1 \le i \le N} \varphi_{t-1}(i)a_{i,j}$$

Obtención De Estados:

$$\widehat{q_T} = \arg\max_{1 \le i \le N} \varphi_T(i)$$

$$\widehat{q_t} = \psi_{t+1}(\widehat{q_{t+1}}) \qquad t = T-1, \dots, 1$$

Segunda pregunta resuelta.

Baum-Welch – Estimación De Modelos

$$\mathbb{Y} = \{y_1, y_2, \dots, y_T\} \longrightarrow M = (a_{i,j}, \pi_i, b_j)$$

Vamos a tomar las distribuciones como normales (o mezclas de normales):

$$b_j(y_t) = N(y_t | \mu_j, \Sigma_j)$$

Como en EM, la función gamma hace de función de responsabilidad:

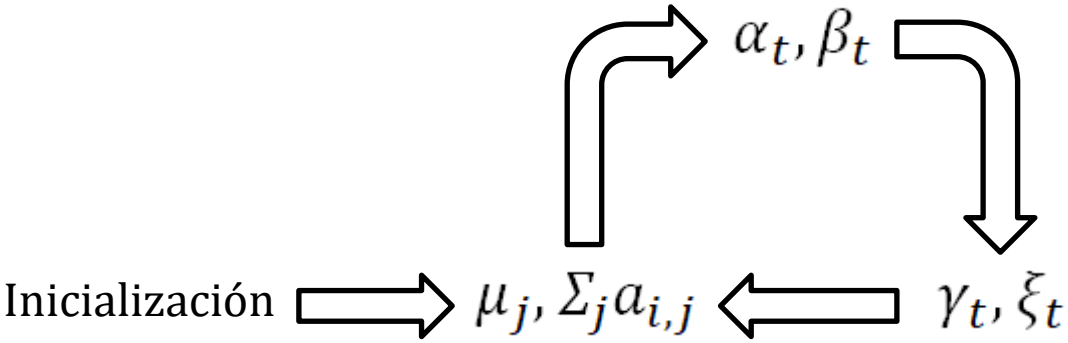
$$\mu_j = \frac{\sum_{t=1}^{T} \gamma_t(j) y_t}{\sum_{t=1}^{T} \gamma_t(j)}$$

$$\Sigma_j = \frac{\sum_{t=1}^{T} \gamma_t(j)(y_t - \mu_j)(y_t - \mu_j)^T}{\sum_{t=1}^{T} \gamma_t(j)}$$

$$a_{i,j} = \frac{\sum_{t=1}^{T} \xi_t(i,j)}{\sum_{t=2}^{T} \gamma_t(j)}$$

Baum-Welch – Estimación De Modelos

También es un proceso iterativo:



Se repite hasta que $P(\mathbb{Y}|M)$ no se incrementa mas.

Tercera pregunta resuelta.

Modelo De Lenguaje

Reconocimiento De Frases w:

$$\hat{w} = \arg\max_{w} P(w|\mathbb{Y}) = \arg\max_{w} P(\mathbb{Y}|w)P(w)$$

$P(\mathbb{Y}|w) \equiv P(\mathbb{Y}|M) \longrightarrow$ Resuelto con Markov

$P(w) \longrightarrow$ Modelo De Lenguaje

Modelo De Lenguaje

$$w = \{la\ casa\ es\ linda\}$$

$$P(w) \approx \frac{\#\{la\ casa\ es\ linda\}}{\#secuencias} \longrightarrow$$ Mala estimación.

Modelos De Bigramas:

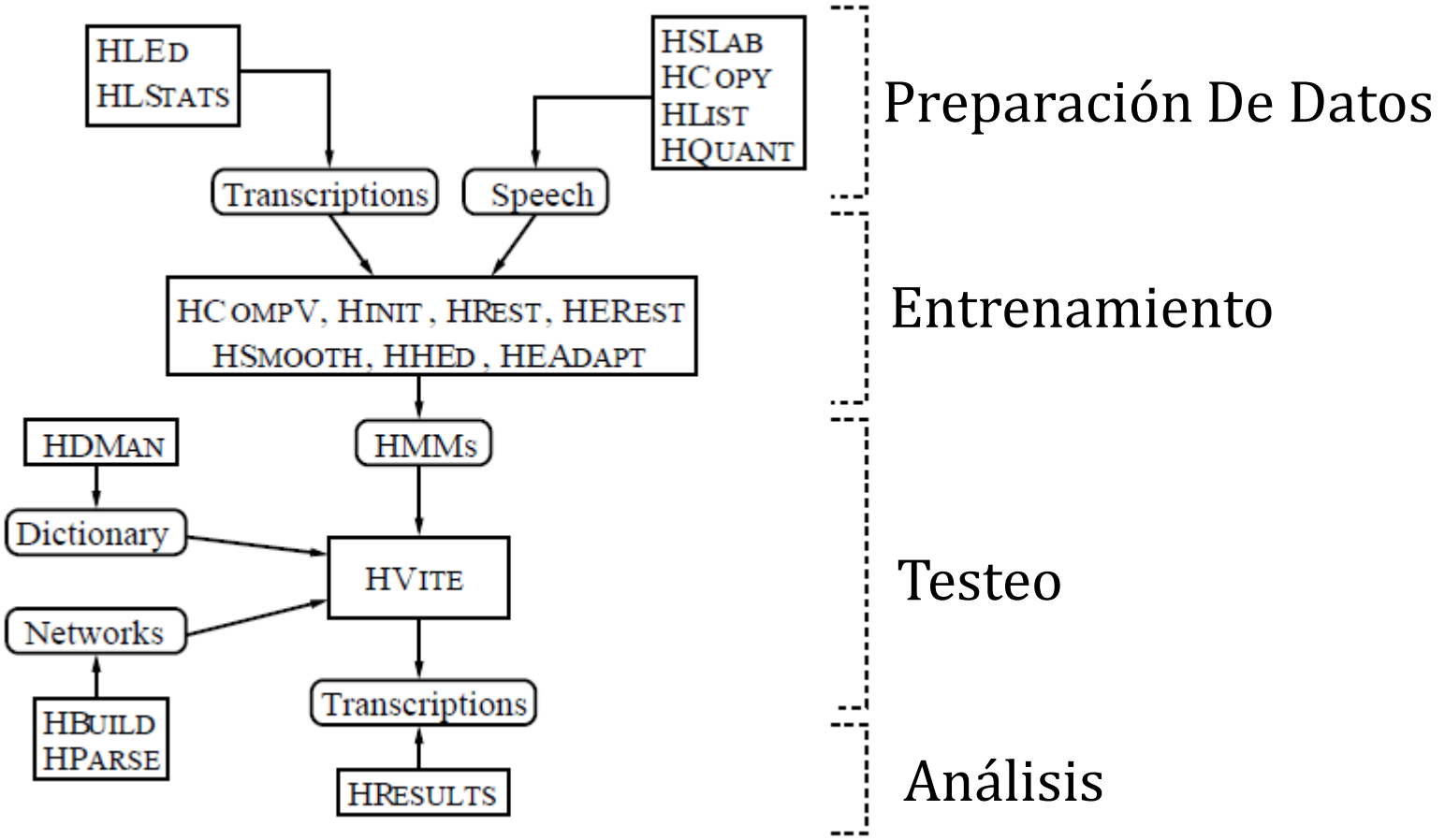
$$P(w) = P(linda|es, casa, la)P(es|casa, la)P(casa|la)P(la)$$

$$P(linda|es, casa, la) \approx P(linda|es) \approx \frac{\#\{es\ linda\}}{\#\{es\}}$$

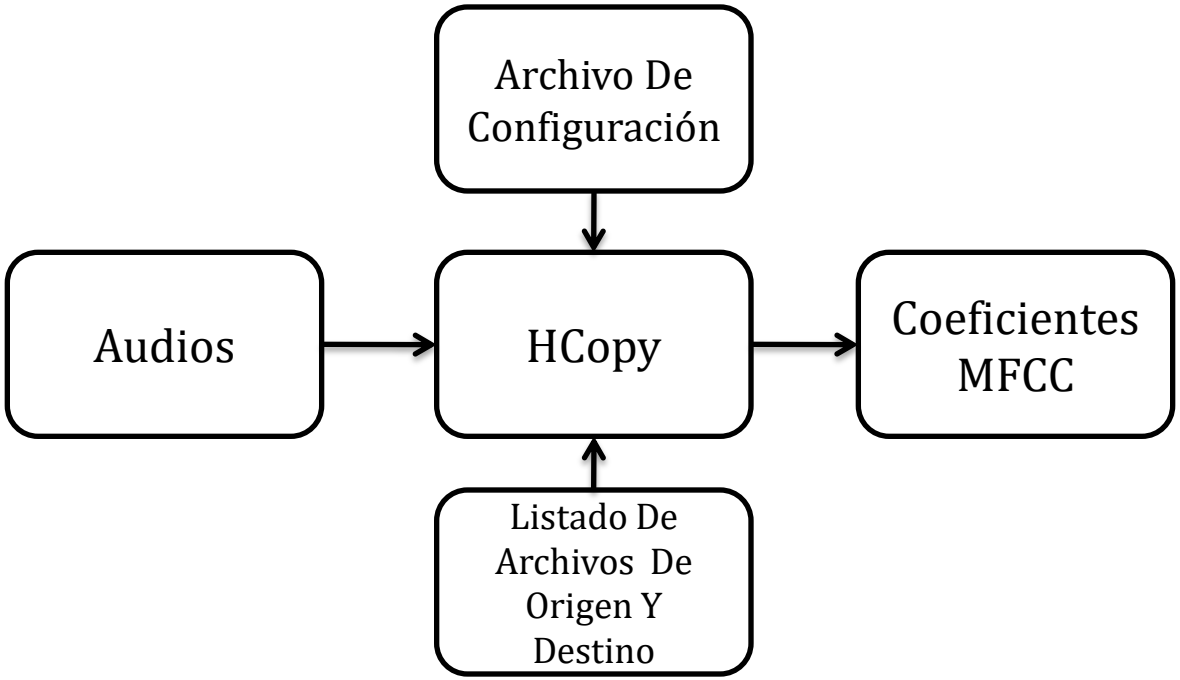
$$P(w) \approx P(linda|es)P(es|casa)P(casa|la)P(la)$$

Hay técnicas mejores y mas avanzadas como los métodos de descuento absoluto y Kneser-Ney

HTK – Trabajo Final

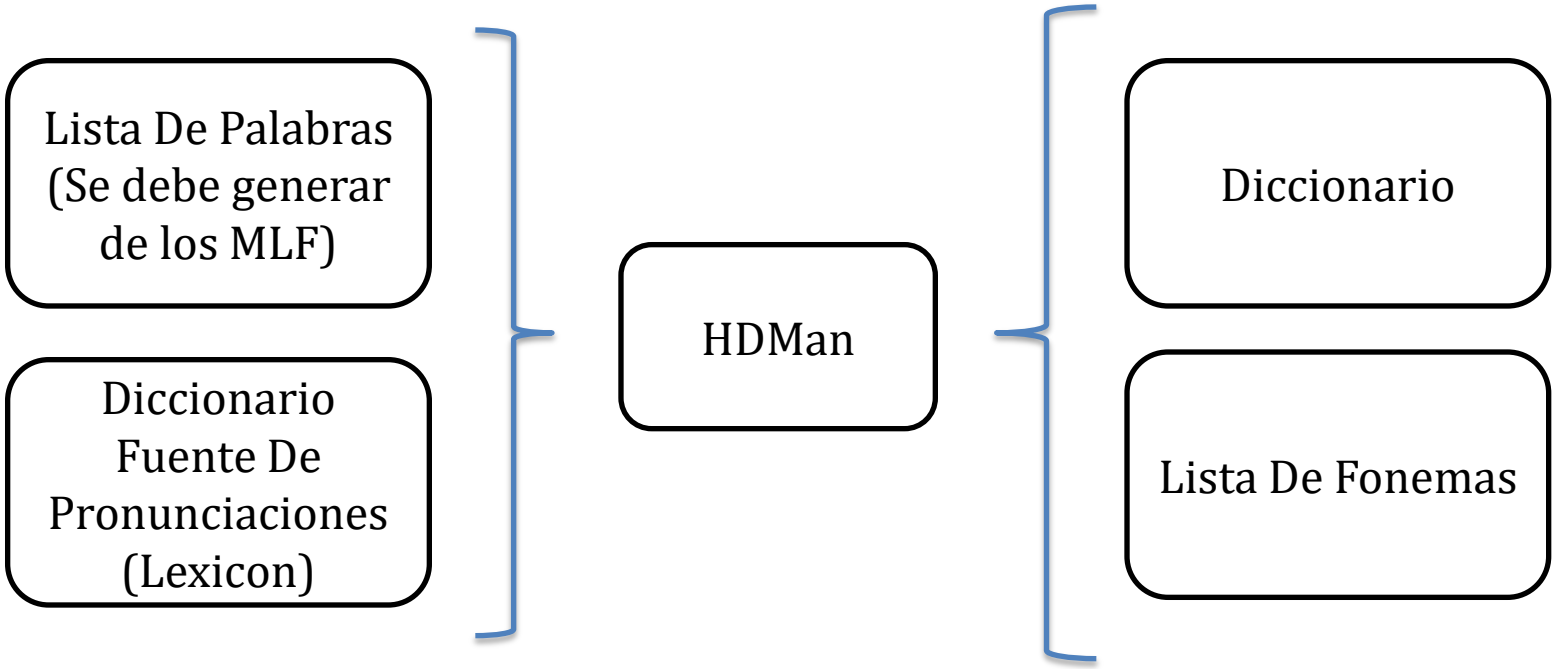


HCopy – Coeficientes MFCC

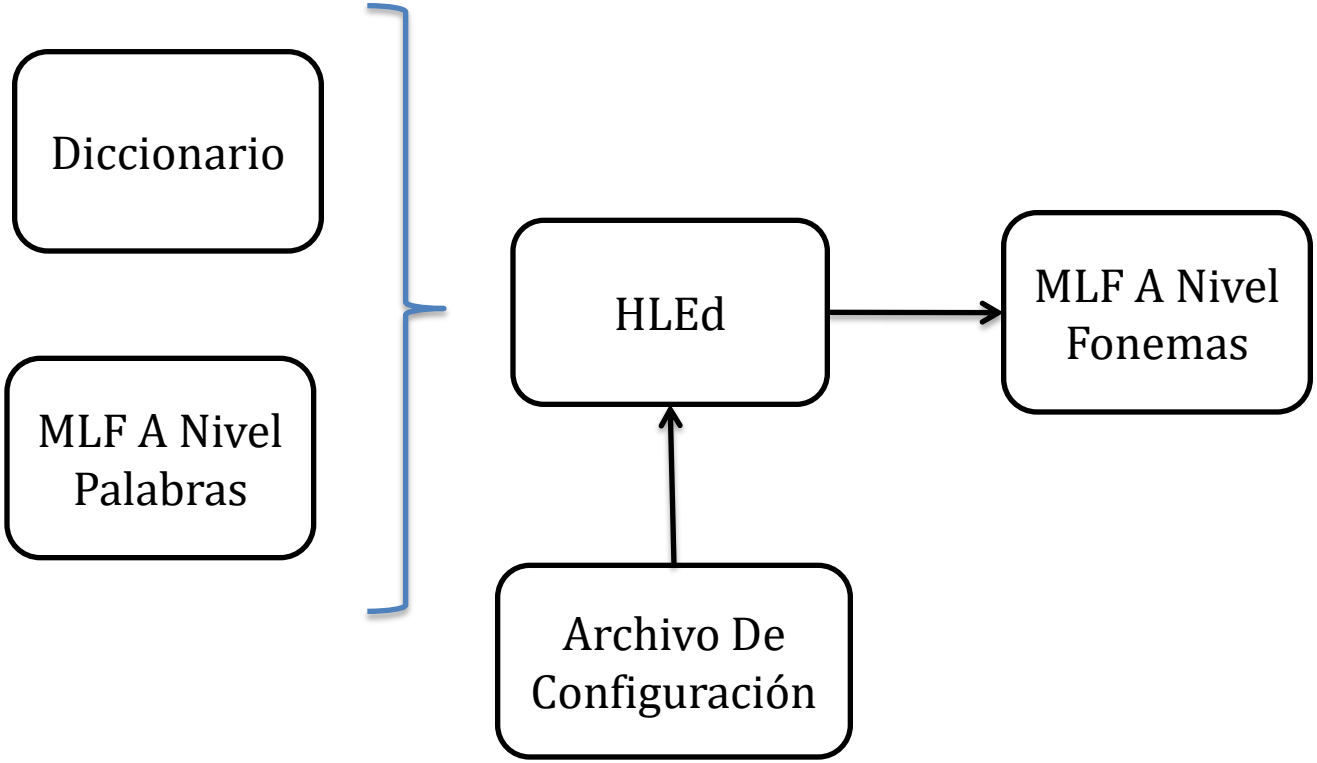


El archivo de configuración contiene información de los formatos de origen y destino (MFCC), la forma de ventanear, la frecuencia de muestreo y el numero y tipo de coeficientes MFCC.

HDMAN – Generación De Diccionarios

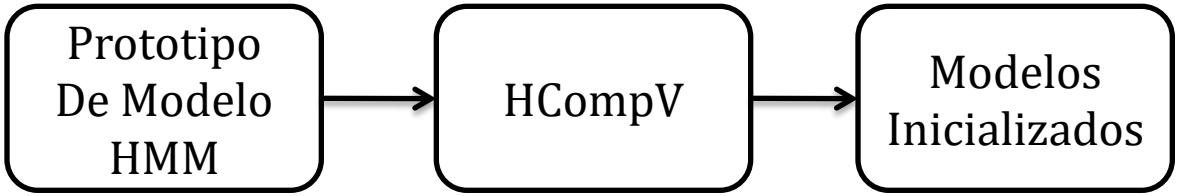


HLEd – Generación De MLF A Nivel Fonemas



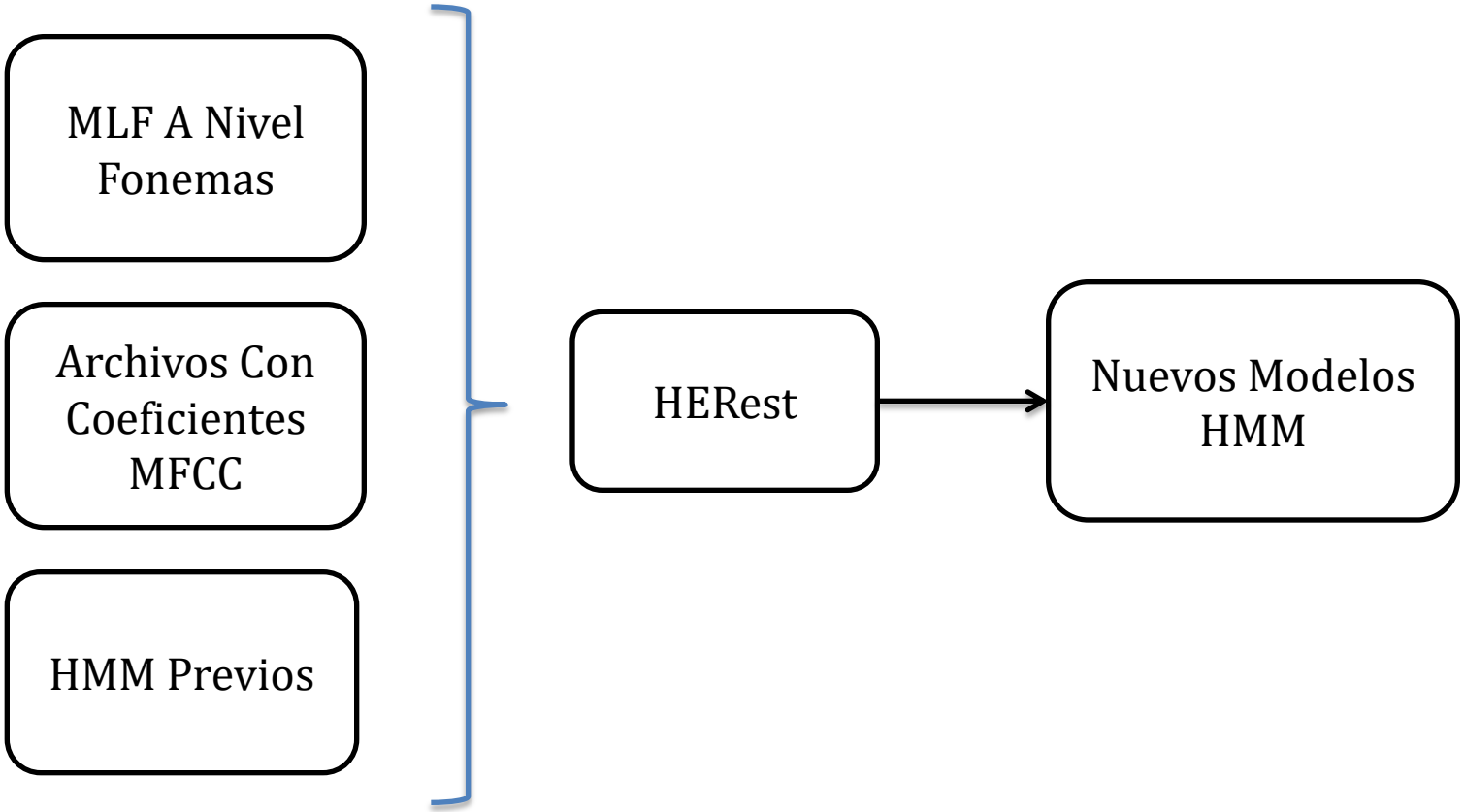
En el archivo de configuración es donde le decimos que queremos expandir las palabras a fonemas, además podemos indicar que agregue silencios al principio y final.

HCompV – Inicialización De Modelos De Fonemas



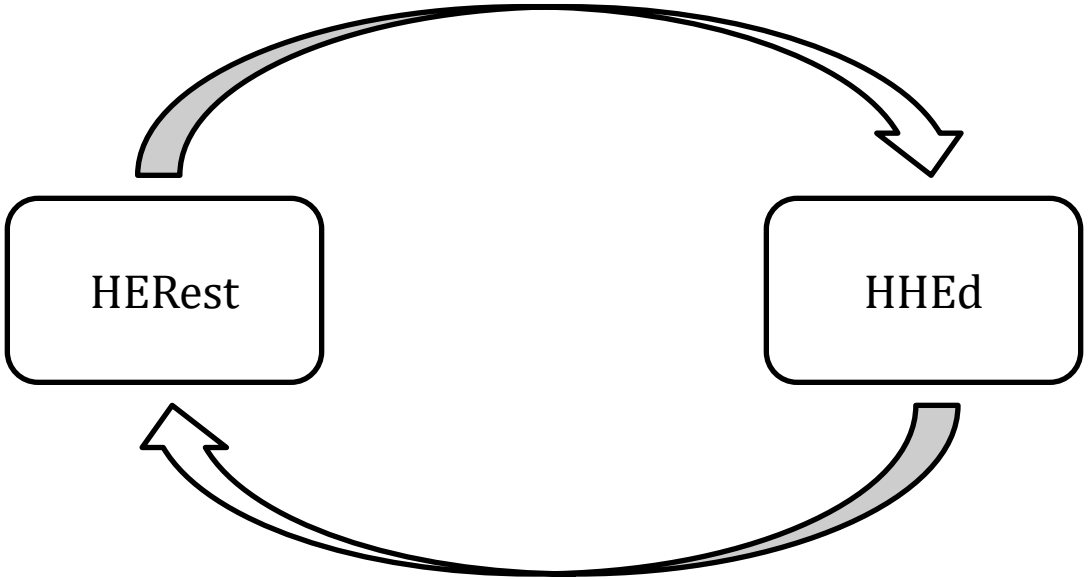
Computa medias y varianzas globales y las asigna a los modelos.

HERest – Baum-Welch



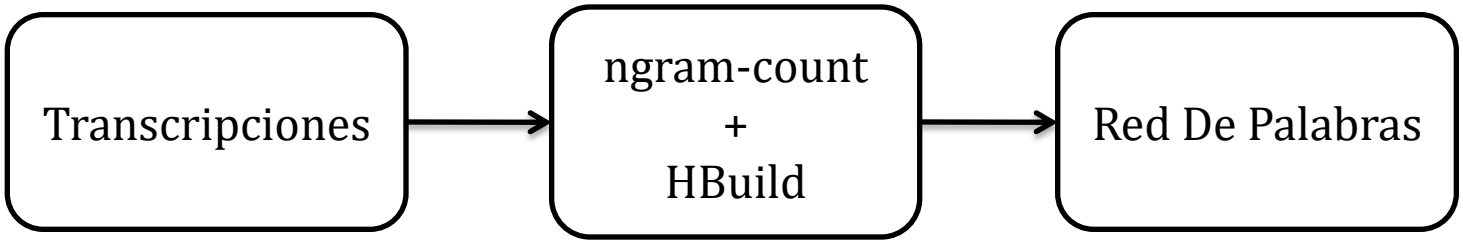
HHEd – Incremento De Gaussianas

Vamos incrementando el numero de gaussianas por estado con HHEd.



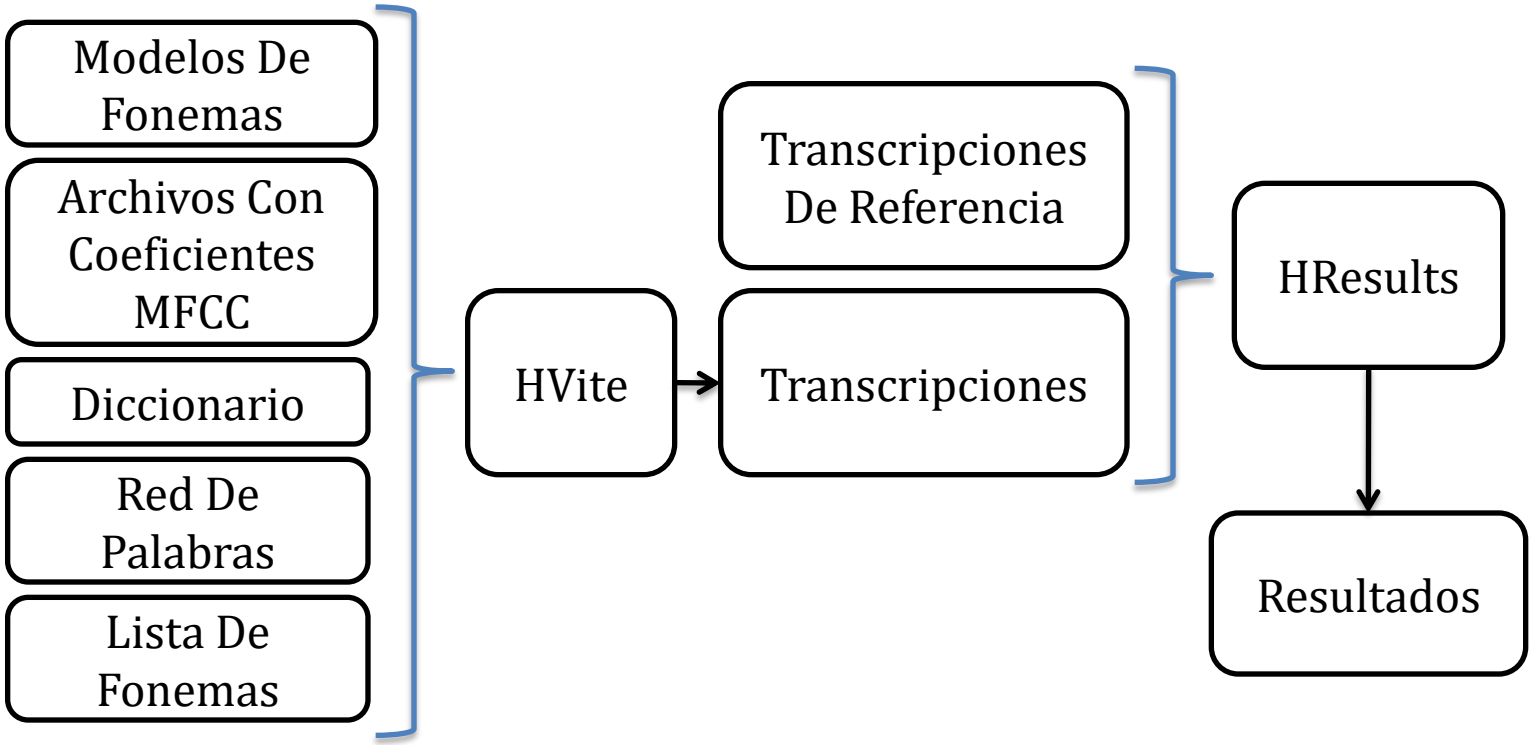
Y vamos generando modelos HMM cada vez mejores.

Modelo De Lenguaje

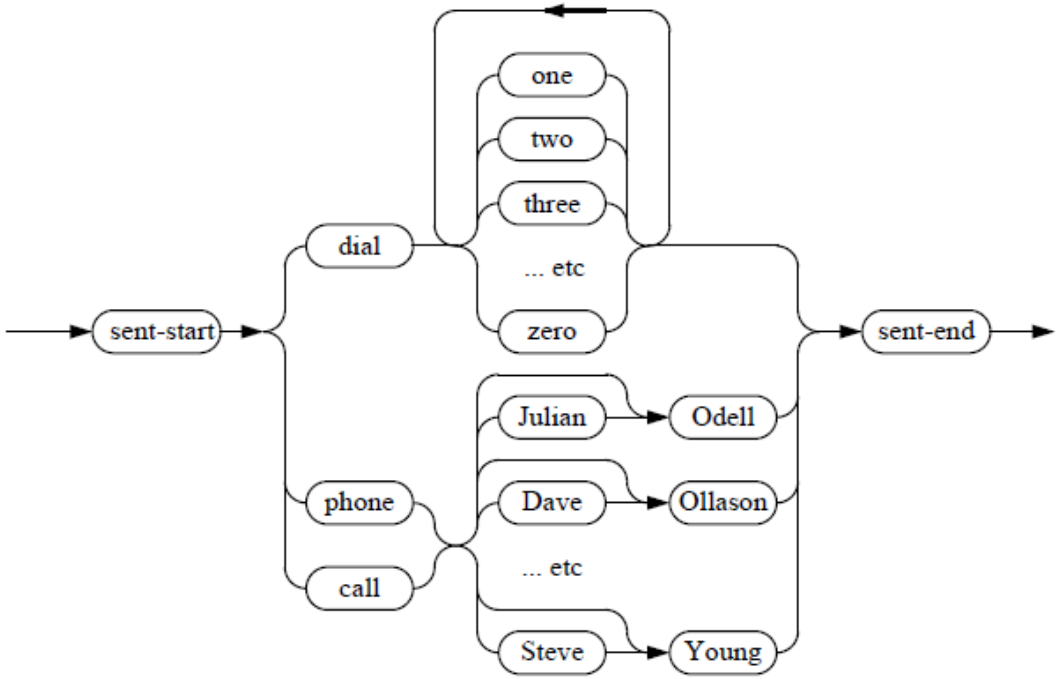


NGRAM-COUNT obtiene los modelos de orden 2 y con HBuild lo convertimos en un formato que el HTK sepa entender (La red de palabras).

HVite + HResults – Reconocimiento



Gramática Finita




```
$digit = ONE | TWO | THREE | FOUR | FIVE |
         SIX | SEVEN | EIGHT | NINE | OH | ZERO;
$name  = [ JOOP ] JANSEN |
         [ JULIAN ] ODELL |
         [ DAVE ] OLLASON |
         [ PHIL ] WOODLAND |
         [ STEVE ] YOUNG;
( SENT-START ( DIAL <$digit> | (PHONE|CALL) $name) SENT-END )
```

Secuencia Del TP

- Generar el archivo con la gramática.
- Generar la red de palabras con HParse (Será necesario para el HVite).
- Generar el diccionario con HDMan.
- Generar las frases con HSGen.
- Grabar audios.
- Generar coeficientes MFCC con HCopy.
- Reconocimiento con HVite + HResults.