Equivalencias entre las etiquetas UAM y EHU

Luis Javier Rodríguez Fuentes
Departamento de Electricidad y Electrónica
Facultad de Ciencias. Universidad del País Vasco
Apartado 644. 48080 Bilbao. SPAIN
e-mail: luisja@we.lc.ehu.es

7 de mayo de 2002

A continuación se muestran las reglas de transformación de las etiquetas UAM para generar las equivalentes en formato simplificado EHU:

- <id_locutor> $\longrightarrow$ id_locutor[índice_turno]: ,

  donde *id_locutor* representa las distintas identidades que puede tomar un locutor, que se traducen como sigue:

Traducción	**Tipo de interlocutor**
HN $\longrightarrow$ HN	hablantes o interlocutores principales
HL $\longrightarrow$ HL	hablantes o interlocutores secundarios
Encuestador $\longrightarrow$ HE	la persona que formula las preguntas en cierto tipo de entrevistas de tipo encuesta
Encuestado N $\longrightarrow$ HeN	las personas que responden a las preguntas en cierto tipo de entrevistas de tipo encuesta
Público $\longrightarrow$ HP	el público en grabaciones de programas de radio o televisión
Todos $\longrightarrow$ HT	todos los hablantes simultáneamente
no identificado $\longrightarrow$ HX	un hablante sin identificar (puede ser alguno de los hablantes principales u otro que no haya intervenido hasta entonces)

  En la tabla anterior N representa un número natural (1, 2, 3, etc.) y L una letra del alfabeto (a, b, c, etc.).

- : $\longrightarrow$,
  ; $\longrightarrow$,
  ... (aislados) $\longrightarrow \varepsilon$
  " $\longrightarrow \varepsilon$,

  donde ε representa la cadena vacía.

- X<palabra cortada> $\longrightarrow$ (lu Y X)
  X<palabra cortada><sic> $\longrightarrow$ (lu Y X) ,

  donde X es la palabra cortada e Y la palabra completa. En primera instancia el transcriptor automático hace Y=X, ya que desconoce cuál debería ser la palabra completa. El anotador será quien deba corregir el valor de Y. Por otro lado, la marca *<sic>* que sigue a algunas palabras cortadas incide en el hecho de que fue el locutor quien realmente utilizó esa forma en lugar de la correcta.

- X<(Y)>Z ⟶ (lm XYZ XZ)
  X<(Y)>Z<sic> ⟶ (lm XYZ XZ) ,

  donde Y representa un fonema borrado, XZ la palabra tal cual fue dicha y XYZ la palabra tal como tendría que decirse. La partícula *<sic>* aparece en ocasiones con la misma funcionalidad descrita anteriormente.

- X<sic> ⟶ (lm X X) ,

  donde la partícula *<sic>*, como en casos anteriores, indica que la palabra X no mantiene la concordancia o ha sido pronunciada de manera anómala.

- <siglas> X </siglas> ⟶ (ls X) ,

  donde estamos suponiendo que la palabra X no se lee de forma normal, sino deletreándola. En los casos en que las siglas se lean (como por ejemplo en *CIA*), el anotador se verá obligado a eliminar la marca.

- <extranjero> X </extranjero> ⟶ (lx X Y) ,

  donde X es la palabra en su ortografía original e Y la palabra según se pronuncia en castellano. En primera instancia el transcriptor automático hace Y=X, ya que desconoce cuál es la pronunciación en castellano de X. Por tanto, el anotador deberá corregir el valor inicial de Y para que encaje con la pronunciación de dicha palabra en castellano (en realidad, con la pronunciación que el locutor haya realizado).

- <vacilación> ⟶ (fb)

- Xv... ⟶ X(a v) (p) , si v es una vocal o alguna de las consonantes *n*, *l* o *s*.

- Xc... ⟶ Xc (p) , si c es una consonante distinta a *n*, *l* o *s*.

- <silencio> ⟶ (p)

- <fático=afirmación> ⟶ (lg sí)
  <fático=negación> ⟶ (lg no)
  <fático=interrogación> ⟶ (fm)
  <fático=duda> ⟶ (fm)
  <fático=exclamación> ⟶ (fa)
  <fático=admiración> ⟶ (fa)
  <fático=asombro> ⟶ (fa)
  <fático=sorpresa> ⟶ (fa)
  <fático=orden> ⟶ (fa) ,

  teniendo en cuenta que la realización acústica de las cinco últimas categorías es muy variable, por lo que el anotador se verá obligado casi siempre a corregir la transcripción por defecto *(fa)* de las mismas.

- eh... ⟶ (fe)

- <ininteligible> ⟶ [NO TRANSCRITO]
  <texto no transcrito> ⟶ [NO TRANSCRITO]

- <borrado involuntario> X </borrado involuntario> ⟶ [CORTE]

- <texto leído> $\longrightarrow \varepsilon$
  </texto leído> $\longrightarrow \varepsilon$
  <cantando> $\longrightarrow \varepsilon$
  </cantando> $\longrightarrow \varepsilon$
  <onomatopéyico> $\longrightarrow \varepsilon$
  </onomatopéyico> $\longrightarrow \varepsilon$
  <argot> $\longrightarrow \varepsilon$
  </argot> $\longrightarrow \varepsilon$
- <simultáneo> X + fin de turno $\longrightarrow$ (o X)
  inicio de turno + X </simultáneo> $\longrightarrow$ (o X)
- <tos> $\longrightarrow$ (nt)
  <carraspeo> $\longrightarrow$ (nt)
- <X> $\longrightarrow$ (nw) ,

  donde cualquier otra marca X no anotada anteriormente se interpreta como un ruido externo (aplausos, risas, música, etc.)

La cabecera del formato UAM será eliminada en el formato EHU. El resultado de esta transformación será tomado como punto de partida (transcripción original) por el anotador.

Apéndice 3. Script de conversión UAM $\longrightarrow$ EHU.

```
#!/usr/bin/perl -w

#        Procesa la entrada standard y entrega el resultado en la salida standard
#        Entrada: transcripción de un diálogo en formato UAM
#        Salida:  transcripción de un diálogo en formato simplificado EHU

#        secuencias alfanuméricas de longitud 0 o mayor
        $w0 = "([a-zA-Z0-9ñyÑYáéíóúÁÉÍÓÚ]*)";

#        secuencias alfanuméricas de longitud 1 o mayor
        $w1 = "([a-zA-Z0-9ñyÑYáéíóúÁÉÍÓÚ]+)";

while (<>)
{
#        Saltamos todo hasta encontrar la marca de inicio de transcripción
        last if /<texto>/;
}

#        Concatenamos las líneas correspondientes al turno $indiceturno en
#        el array de caracteres $turno[$indiceturno]
$indiceturno = 0;

while (<>)
{
#        Terminamos cuando vemos la marca de fin de transcripción
        last if /<\/texto>/;

#        Se saltan las líneas vacías
        next if /^[ ]*$/;

#        Primera línea de un turno con un hablante normal: H1, H2, etc.
        if (/^[ ]*<H(.*?)>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*?<H(.*?)>(.*)\n/;
                $turno[$indiceturno] = "H$1\[$indiceturno\] $2 ";
        }
#        Primera línea de un turno con un encuestador
        else{
        if (/^[ ]*<[Ee]ncuestador>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*<[Ee]ncuestador>(.*)\n/;
                $turno[$indiceturno] = "HE\[$indiceturno\] $1 ";
        }
#        Primera línea de un turno con un encuestado
        else{
        if (/^[ ]*<[Ee]ncuestado[ ]+([0-9]+)>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*<[Ee]ncuestado[ ]+([0-9]+)>(.*)\n/;
                $turno[$indiceturno] = "He$1\[$indiceturno\] $2 ";
        }
#        Primera línea de un turno con un hablante no identificado
        else{
        if (/^[ ]*<no identificado>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*<no identificado>(.*)\n/;
                $turno[$indiceturno] = "HX\[$indiceturno\] $1 ";
        }
#        Primera línea de un turno en el que interviene el público
        else{
        if (/^[ ]*<público>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*<público>(.*)\n/;
                $turno[$indiceturno] = "HP\[$indiceturno\] $1 ";
        }
#        Primera línea de un turno en el que intervienen todos los interlocutores
        else{
        if (/^[ ]*<[Tt]odos>(.*)\n/)
        {
                $indiceturno++;
                /^[ ]*<[Tt]odos>(.*)\n/;
                $turno[$indiceturno] = "HT\[$indiceturno\] $1 ";
        }
#        Cualquier otra línea del turno
        else
        {
                /(.*)\n/;
                $turno[$indiceturno] = $turno[$indiceturno] . "$1 ";
        }}}}}}
}
```

```
#        Ahora procesamos cada turno por separado, desde $i=1 hasta $i=$indiceturno
$overlap=0;

for ($i=1; $i<=$indiceturno; $i++)
{
#        Antes de procesar nada, marcamos los segmentos de transcripción cuya señal
#        se ha perdido con la etiqueta [CORTE], y los eliminamos
        $turno[$i] =~ s/<borrado involuntario>.*?<\/borrado involuntario>/[CORTE]/g;

#        Sustituimos punto_y_comas(;) y dos_puntos(:) por comas (,)
#        Al mismo tiempo, a cada coma se le añade un espacio por delante y por detrás
        $turno[$i] =~ s/[,;:]/ , /g;

#        En este momento, una vez filtrados los dos_puntos, marcamos el inicio del turno
#        precisamente con dos_puntos
        $turno[$i] =~ s/^(H.*?\[[0-9]+\])/$1: /g;

#        Interrogaciones y admiraciones se rodean de espacios también
        $turno[$i] =~ s/([¿?¡!])/ $1 /g;

#        Sustituimos las dobles comillas por espacios
        $turno[$i] =~ s/"/ /g;

#        Las secuencias de dos o más espacios son sustituidas por un solo espacio
        $turno[$i] =~ s/[ ]+/ /g;

#        Algunas marcas <( )> se han encontrado cambiadas: (< >)
        $turno[$i] =~ s/\(<$w1>\)/<\($1\)>/g;

#        Los paréntesis que no forman parte de la marca "<( )>"
#        son sustituidos por comas y rodeados por espacios blancos
        $turno[$i] =~ s/([^<])\(/$1 , /g;
        $turno[$i] =~ s/\)([^>])/ , $1/g;

#        Los sonidos borrados se transcriben como palabras mal pronunciadas
#        En las transcripciones UAM a veces estas palabras van seguidas de <sic> (redundante)
        $turno[$i] =~ s/ $w0<\($w1\)>$w1<\($w1\)>$w0( |\.|<sic>)/ \(lm $1$2$3$4$5 $1$3$5\) $6/g;
        $turno[$i] =~ s/ $w0<\($w1\)><\($w1\)><\($w1\)>$w0( |\.|<sic>)/ \(lm $1$2$3$4$5 $1$5\) $6/g;
        $turno[$i] =~ s/ $w0<\($w1\)><\($w1\)>$w0( |\.|<sic>)/ \(lm $1$2$3$4 $1$4\) $5/g;
        $turno[$i] =~ s/ $w0<\($w1\)>$w0( |\.|<sic>)/ \(lm $1$2$3 $1$3\) $4/g;

#        La etiqueta <sic> suele marcar una palabra mal pronunciada o sin concordancia
        $turno[$i] =~ s/$w1( | , )?<sic>/\(lm $1 $1\)/g;
#        Si después de esto queda algún <sic>, será redundante y lo eliminamos
        $turno[$i] =~ s/<sic>//g;

#        En formato UAM los segmentos simultáneos empiezan al final de un turno...
        if ($overlap==0 && $turno[$i] =~ /<simultáneo>/)
        {
                $overlap=1;
                $turno[$i] =~ s/<simultáneo>(.*)\.[ ]*$/\(o $1\) \./g;
                $turno[$i] =~ s/<simultáneo>(.*)$/\(o $1\) \./g;
        }
#        ...y terminan al comienzo del siguiente
        else
        {
                if ($overlap==1 && $turno[$i] =~ /<[\/]?simultáneo>/)
                {
                        $overlap=0;
                        $turno[$i] =~ s/^(.*?): (.*)<[\/]?simultáneo>/$1: \(o $2\)/g;
                }
        }
#        En ambos casos parentizamos el segmento solapado con la marca "o"

#        Hay marcas que no transcribimos, simplemente las borramos
        $turno[$i] =~ s/<texto leído>//g;
        $turno[$i] =~ s/<\/texto leído>//g;
        $turno[$i] =~ s/<cantando>//g;
        $turno[$i] =~ s/<\/cantando>//g;
        $turno[$i] =~ s/<onomatopéyico>//g;
        $turno[$i] =~ s/<\/onomatopéyico>//g;
        $turno[$i] =~ s/<argot>//g;
        $turno[$i] =~ s/<\/argot>//g;

#        Los segmentos de señal no transcritos o ininteligibles los marcamos con la etiqueta [NO TRANSCRITO]
        $turno[$i] =~ s/<ininteligible>/[NO TRANSCRITO]/g;
        $turno[$i] =~ s/<texto no transcrito>/[NO TRANSCRITO]/g;

#        Las secuencias " eh... " y similares (por ejemplo, " eh " cuando no va entre
#        interrogaciones, pero no así " e<palabra cortada> ") se transcriben como pausas
#        habladas, es decir, como "(fe)"
        $turno[$i] =~ s/([^ ¿])([ ]+)(E|e)h?(\.*)([ ]+)([^ ?]|$)/$1$2\(fe\)$5$6/g;

#        Cuando una palabra termina en vocal, o en ciertas consonantes (n,l,s), y va
#        seguida de puntos suspensivos, por ejemplo, "el...", la vocal o consonante
```

```
#        se transcribe alargada y seguida de una pausa, es decir, "e(a l) (p)"
       $turno[$i] =~ s/$w0([aeiyounlsáéíóú])\.{2,}?/$1\(a $2\) \(p\) /g;
       $turno[$i] =~ s/$w0([^aeiyounlsáéíóú])\.{2,}?/$1$2 \(p\) /g;

#        Las vacilaciones se transcriben como pausas habladas pero sin asignarles una
#        identidad fonética
       $turno[$i] =~ s/<vacilación>/\(fb\)/g;

#        Los silencios y los puntos suspensivos se transcriben como pausas
       $turno[$i] =~ s/<silencio>|\.\.\./\(p\)/g;

#        Las palabras cortadas se transcriben como tales
       $turno[$i] =~ s/$w1<palabra cortada>/\(lu $1 $1\)/g;

#        Las siglas no se pronuncian normalmente sino deletreando; por ello se define
#        una categoría léxica específica: (ls SIGLAS)
       $turno[$i] =~ s/<siglas>(.*?)<\/siglas>/\(ls $1\)/g;

#        Las palabras en lengua extranjera no suelen pronunciarse en castellano; por ello
#        se define una categoría léxica específica: (lx palabra_extranjera pronunciación).
#        Por ejemplo: (lx light lait)
       $turno[$i] =~ s/<extranjero>(.*?)<\/extranjero>/\(lx $1 $1\)/g;

#        Muchos sonidos no forman palabras en sentido estricto, sino que tienen
#        una función fática de afirmación, negación, sorpresa, etc.

#        Las afirmaciones y negaciones guturales se transcriben como entidades léxicas
#        específicas: (lg sí) y (lg no), respectivamente
       $turno[$i] =~ s/<fático=(afirmación|afirmativo)>/\(lg sí\)/g;
       $turno[$i] =~ s/<fático=(negación|negativo)>/\(lg no\)/g;
#        Las categorías "interrogación" y "duda" se pueden transcribir casi siempre
#        como pausas habladas nasalizadas: (fm)
       $turno[$i] =~ s/<fático=(interrogación|interrogativo)>/\(fm\)/g;
       $turno[$i] =~ s/<fático=duda>/\(fm\)/g;
#        El resto de categorías las transcribimos (casi seguro que mal) como (fa)
       $turno[$i] =~ s/<fático=(exclamación|admiración|asombro|sorpresa|orden)>/\(fa\)/g;

#        Toses y carraspeos se transcriben como (nt)
       $turno[$i] =~ s/<tos>|<carraspeo>/ (nt) /g;

#        Lo que queda se transcribe como ruido externo (aplausos, risas, música, etc.)
       $turno[$i] =~ s/<.*?>/ (nw) /g;

#        REGLAS DE FORMATO

#        Todos los turnos finalizan en punto
       $turno[$i] =~ s/(.*)$/$1./g;
#        Varios puntos seguidos se condensan en uno solo
       $turno[$i] =~ s/\.+/\./g;
#        Cada punto se rodea de sendos espacios antes y después
       $turno[$i] =~ s/\./ \. /g;
#        Punto seguido de espacios y coma o punto se transcribe como un único punto
       $turno[$i] =~ s/\.([ ]*[.,])+/\./g;
#        Punto o coma seguido de espacios e interrogación (admiración) se transcribe como
#        interrogación (admiración)
       $turno[$i] =~ s/[.,][ ]*(¿|\?|¡|!)/$1/g;
#        Interrogación (admiración) seguido de espacios y punto o coma se transcribe como
#        interrogación (admiración)
       $turno[$i] =~ s/(¿|\?|¡|!)[ ]*[.,]/$1/g;
#        Coma seguida de espacios y punto se transcribe como punto
       $turno[$i] =~ s/,([ ]*[.])+/\./g;
#        Varias comas seguidas (con posibles espacios intermedios) se funden en una sola
       $turno[$i] =~ s/,([ ]*,)+/,/g;
#        Varios espacios seguidos se funden en uno
       $turno[$i] =~ s/ +/ /g;
#        Si hay un espacio al final del turno, se elimina
       $turno[$i] =~ s/(.*) $/$1/g;

#        PUNTUACION A TRAVES DE PARENTESIS DE CIERRE
       $turno[$i] =~ s/ [.,] \) \./\) \./g;
       $turno[$i] =~ s/ ([.,]) \)/\) $1/g;
       $turno[$i] =~ s/ (\?|!) \) \./\) $1/g;
       $turno[$i] =~ s/ \)/\)/g;

#        PUNTUACION A TRAVES DE PAUSAS, ALARGAMIENTOS Y RUIDOS
       $turno[$i] =~ s/[.,] ((\((p|f[aemb]|n[wt])\) )+)([.¿?¡!])/$1$4/g;
       $turno[$i] =~ s/([.¿?¡!])(( \((p|f[aemb]|n[wt])\))+) [.,]/$1$2/g;
       $turno[$i] =~ s/, ((\((p|f[aemb]|n[wt])\))+) ,/$1/g;
#        Punto seguido de minúsculas se sustituye por coma
       $turno[$i] =~ s/\. ((\((p|f[aemb]|n[wt])\))+) ([a-záéíóú])/, $1 $4/g;
       $turno[$i] =~ s/\. \($w1 ([a-záéíóú])(.*?)\)/, \($1 $2$3\)/g;
       $turno[$i] =~ s/\. ([a-záéíóú])/, $1/g;

#        Se imprime el turno transcrito a formato EHU, más un salto de línea
       print "$turno[$i]\n";
}
```