Conversor texto a voz en el dialecto venezolano por medio de la concatenación de difonos

Text to speech of the Venezuelan dialect via diphone concatenation

Rodríguez, Manuel
Departamento de Electrónica y Comunicaciones, Facultad de Ingeniería, Universidad de Los Andes

Mora, Elsa
Departamento de Lingüística, Facultad de Humanidades y Educación, Universidad de Los Andes

Recibido: 25-11-2005 Revisado: 10-06-2006

Resumen

En este trabajo presentamos un sistema de síntesis de habla con voz venezolana a partir de texto español. Este sistema está basado en el método mbrola de concatenación de difonos para lo cual se utilizó una base de 794 difonos (Rodríguez et al. 2003) que permite generar cualquier enunciado en español venezolano. Se describe en detalle el módulo de conversión ortográfico fonético escrito en lenguaje Perl. Complementario a la descripción del módulo se presentan contribuciones en materia de la generación de patrones entonativos simbólicos y numéricos, la duración de los segmentos, cambios fonéticos por contexto y la transformación de la cadena de fonemas a la representación SAMPA.

Palabras clave: tecnologías del habla, síntesis de voz, conversión texto a voz, concatenación de difonos.

Abstract

We give a detailed description of a software module which automatically converts an input text into a file appropriate for speech synthesis using the mbrola speech synthesis program. The system uses the diphone data base for Venezuelan Spanish consisting of 794 diphones (Rodriguez et al 2003). The module is written in Perl. In this way all the necessary ingredients for text to speech are completed. Complementing the software, we include contributions regarding the generation of symbolic and numeric entonation patterns, segment duration, context dependent phonetic changes and the transformation of the phonetic sequence in the SAMPA representation.

Key words: speech technology, speech synthesis, text to speech conversion, diphone concatenation

1 Introducción

En otro trabajo (Rodríguez, Mora, Cave, 2006), se describe el uso de una base de datos (Rodríguez et al 2003) de difonos, vz1, como parte integral de un sistema de síntesis de voz por medio de la concatenación de difonos basado en la técnica de mbrola (Dutoit, 1997). En ese mismo trabajo, se describen los antecedentes de la síntesis de voz (Huang et al 2001), así como el sistema de síntesis de mbrola. En ese sistema, el módulo de entrada tiene la función de convertir un texto en un archivo .pho donde se indica la secuencia de fonos a sintetizar, junto con valores de duración y de entonación de estos fonos. Este archivo se puede armar manualmente por un experto, en función del texto de entrada. En este trabajo, se presenta un módulo de software donde todas las funciones necesarias para crear este archivo se realizan automáticamente. Algunas de estas funciones se describen en base a investigaciones complementarias para definir aspectos importantes como son la generación de patrones entonativos simbólicos y numéricos, la duración de los segmentos, cambios fonéticos por contexto y la transformación de la cadena de fonemas a la representación SAMPA.

En ese trabajo, se justifica el difono como unidad de síntesis. Para el dialecto venezolano del español, se traba-

jó con el conjunto de fonos en su representación SAMPA, propuesto por Mora et al (2001). A continuación se especifica de nuevo este conjunto y en función de su descripción articulatoria:

- Vocales átonas: a, e, i, o, u,
- Vocales tónicas: a*, e*, i*, o*, u*,
- Oclusivas sordas: p, t, k,
- Oclusivas sonoras: b, g, d,
- Fricativas sonoras: B, G, D,
- Africada: tS (correspondiente al "ch" de "chato"),
- Fricativas sordas: f, s, s2 (un alófono en distensión de la "s"), h (sonido de la "j" en "laja"),
- Nasales: m, n, J (J corresponde a la "ñ"),
- Laterales: l, L (corresponde a la "ll" de "llave" y a la "y" de "ayer"), r, rr,
- Glides: j, w (correspondientes respectivamente a las grafemas "i" y "u" en diptongos, como en "quieto" y "jueves"),
- y Pausa: _.

La base de datos de difonos para el español de Venezuela, vz1, se puede bajar de Internet de la página web de mbrola (tcts.fpms.ac.be/synthesis/mbrola.html). Consta de 794 difonos, y tienen una duración total de 120 segundos, es decir, una duración promedio de 150 ms para cada difono.

Por ejemplo, la frase "Este atlas no es étnico.", se convierte en la siguiente secuencia de fonos: "_ e* s2 t e a* D l a s2 n o* e* s2 e* D n i k o _". En la Fig. 1 se muestra el archivo .pho correspondiente. La columna izquierda contiene la secuencia de fonos a sintetizar. Cada fila a su vez contiene los datos acústicos de duración y entonación correspondientes a un fono. Entonces la segunda columna es la duración en milisegundos (ms) del fono, y las columnas siguientes representan valores por pares para la frecuencia fundamental (f0) de vibración de las cuerdas vocales, precisando primero el % del intervalo del fono donde se fija el segundo valor dado en Hz.

Con la información de este archivo, el programa mbrola obtiene del banco de datos de difonos los segmentos requeridos, y además modifica tanto su duración como su entonación en función de los requerimientos de la frase, y los va concatenando, creando así un archivo de voz, hasta terminar la frase.

2 Conversión texto a voz para el dialecto venezolano

Se ha desarrollado un programa en lenguaje Perl para la conversión automática de texto ortográfico a un archivo .pho para la síntesis con el método mbrola del dialecto venezolano del español, que utiliza la base de datos vz1 de los difonos venezolanos descritos anteriormente.

El lenguaje Perl es particularmente ágil y apto para procesado de texto. El módulo fue desarrollado y probado para el sistema operativo linux, y con unas pequeñas adaptaciones trabaja también en varios ambientes Win

Fonema	ms	% f0	% f0
_	50	0 120	
e*	108	0 100	30 130
s2	110		
t	85		
e	90		
a*	108	0 100	30 130
D	60		
l	80		
a	90		
s2	110		
n	80	0 100	
o*	108	30 130	
e*	108	0 100	30 130
s2	110		
e*	108	0 100	30 130
D	60		
n	80		
i	80		
k	100		
o	90	99 80	
_	250	99 80	

Fig. 1. Un ejemplo de un archivo .pho, para la frase "Este atlas no es étnico."

dows, incluyendo Windows98, WindowsNT y WindowsXP. El programa nuestro es una modificación de un programa escrito por Alistair Conkie, ttp.pl, para el español peninsular, también disponible en la página web de mbrola.

Podemos dividir el trabajo realizado en las siguientes partes principales: a) preprocesado del texto, b) procesado lingüístico, y c) generación de patrones entonativos.

2.1 Preprocesado del texto

En el preprocesado del texto se trata de convertir cualquier símbolo o caracter especial en una secuencia de palabras enteras. Entre estos se encuentra

- Los números Romanos, de la I a la MMMCMXCIX.
- Los números enteros desde 0 hasta 999.999.
- Signos incluyendo $, €, %, +, -, @
- Direcciones electrónicas
- Siglas y acrónimos, sobre todo de uso común en el que hacer venezolano. Aquí vale mencionar el tratamiento diferenciado dependiendo de las siglas a sintetizar, de modo que la "UCV" se deletrea, la "ULA" se maneja como una palabra normal, y la "CANTV" tiene una

parte que se pronuncia como sílaba, "CAN", y lo demás que se deletrea, "te ve".
- Símbolos de productos, por ejemplo el rifle "AK47" que debe transcribirse como " a ka cuarenta y siete".
- Abreviaciones incluyendo Sra., Dr., Bs., etc., que se pronuncian "señora", "doctor", "bolívares", "etcétera", respectivamente.

2.2 El procesado lingüístico

Esta parte incluye un módulo de transcripción fonética, como también de acentuación y de silabificación.

2.2.1 La trascripción ortográfica fonética

Las transcripciones grafema a fonema son importantes porque:
- es ineficiente almacenar las transcripciones de todas las palabras posibles
- es imposible guardar todos los nombres propios, topónimos, ...
- constantemente aparecen neologismos
- hay que poder convertir las palabras extranjeras de uso habitual
- el sistema tiene que ser robusto a posibles errores tipográficos

La trascripción ortográfica fonética se realiza por medio de sustituciones, como se indica en la Tabla 1. Se realiza en 2 pasos para evitar errores o confusiones al haber más de un cambio en una secuencia de 2 o 3 letras. Estas transformaciones se realizan al inicio del programa. En el primer paso por el texto, se atiende grafemas con alternativas de trascripción (c, g, y) y se elimina la u muda. En el segundo paso por el texto, se atiende los demás grafemas, con atención especial a los diptongos y triptongos. Más adelante se describirán otras transformaciones que se realizan al final del programa

Tabla 1. Transcripciones ortográficas fonéticas iniciales

Secuencia Original	Paso I (venezuelan_rules1)	Paso II (venezuelan_rules2)
ce	se	
ci	si	
c'e	s'e	
c'i	s'i	
ge	je	
g'e	j'e	
gi	ji	
g'i	j'i	
gue	ge	
gui	gi	
gu'e	g'e	
gu'i	g'i	
que	ke	
qui	ki	
qu'e	k'e	
qu'i	k'i	
q	k	
x	ks	
ya	lla	
ye	lle	
yi	lli	
yo	llo	
yu	llu	
y'a	ll'a	
y'e	ll'e	
y'i	ll'i	
y'o	ll'o	
y'u	ll'u	
	ia	j_a
	ie	j_e
	io	j_o
	iu	j_u
	ua	W_a
	ue	W_e
	ui	W_i
	uo	W_o
	i'a	j_a:
	i'e	j_e:
	i'o	j_o:
	i'u	j_u:
	u'a	W_a:
	u'e	W_e:
	u'i	W_i:
	u'o	W_o:
	ai	a_j
	ay	a_j
	au	a_W
	ei	e_j
	ey	e_j
	eu	e_W
	oi	o_j
	oy	o_j
	y	i
	i'ai	j_a:_j
	i'ei	j_e:_j
	v	b
	c	k
	ch	c
	hue	w_e
	hie	y_e
	h	\-
	j	x
	ll	L
	~n	~
	rr	R
	w	w
	z	S

2.2.2 La acentuación

En cuanto a la acentuación, la mayoría de palabras en español lleva un acento; a su vez, muchas se marcan con el acento ortográfico. Entre las palabras con acento pero que no llevan acento ortográfico tenemos los de acento agudo (en la penúltima sílaba) que terminan en vocal, "s" o "n" y los de acento grave (en la última sílaba), que terminan con cualquier otra letra. Para el sintetizador, este acento se tiene que colocar en la transcripción fonética. Hay palabras excepcionales que no llevan acento:

De una sílaba: a, al, aun, vos, de, del, e, el, en, quien, con, cual, cuan, la, las, los, le, les, lo, mas, me, mi, mis, ni, no, nos, o, os, por, pos, pues, que, quien, se, si, sin, so, su, sus, te, tan, tu, tras, tus, u, un, y; *Mas de una Silaba*: ante, aunque, bajo, cabe, casi, cerca, como, contra, cuando, cuanto, desde, donde, entre, excepto, frente, hace, hacia, hasta, menos, mientras, para, pero, porque, puesto, que, salvo, segun, sino, sobre, una. Y hay otro grupo de excepciones a la norma de un acento por palabra que son las palabras que terminan en *mente*, y que llevan dos acentos; a su vez, a este grupo hay las excep-

ciones de “lamente”, “demente” y “tormente”, que tienen un acento solamente.

2.2.3 La silabificación

En cuanto a la silabificación, se utilizan las reglas de Van Gerwen, las cuales se aplican en forma secuencial y son las siguientes:

- Si hay un grupo consonántico de (bfgkpt) seguido por l, o de (bdfgkpt) seguido por r, entonces aquí empieza una sílaba.
- Si tenemos una consonante (incluyendo h muda) seguida por una vocal o glide, entonces aquí empieza una sílaba.
- Si una palabra empieza por vocal-consonante-vocal (VCV), entonces la primera vocal forma sílaba.
- Si hay dos vocales seguidas (que no sean glides), entonces entre ellas hay una división silábica.

Aquí también hay excepciones. Hay un grupo de palabras con grupos consonánticos donde las sílabas se dividen “internamente”: sub(lev, lin, lun, ray, rei, rep, rog).

2.3 Generación de patrones entonativos

La entonación es una parte esencial del habla (Quilis 1983) y “…está relacionada básicamente con la percepción, a lo largo de un enunciado, de los cambios de frecuencia de vibración de las cuerdas vocales (f0). Esos cambios crean la melodía del discurso…”(Mora 1998:43). El programa establece la función de entonación, f0, en dos partes: primero hace una asignación simbólica, utilizando una adaptación de una representación propuesta originalmente por Pierrehumbert (1980) en base a la teoría métrica autosegmental de la entonación, y modificada posteriormente a la representación ToBI (para mayor información consultar el capítulo de Sosa en Prieto 2003), y después, en función de esta representación, asigna una entonación numérica a fonemas puntuales. Finalmente, entre estos valores puntuales, el programa mbrola realiza una interpolación lineal.

Tabla 2. Asignación de valores simbólicos ToBI a la entonación.

Condición	Valor simbólico de entonación
Pausa inicial de frase	H[<1.0>
“Minor Phrase Boundary”, es decir, frontera entre palabra tipo CW y FW	L-H]<0.2>
':' o '-'	L-H]<0.7>
';'	L-L]<0.7>
'.'	L-L]<1.0>
'!'	L-L]<1.0>
'?'	L-L]<1.0>
',' o '(' o '[' o '{' o ')' o ']' o '}' o '"'	L-H]<0.7>
Vocal tónica	H*<0.5>

Pierrehumbert se centra en una descripción fonológica de la entonación, y para lo cual hace uso de dos niveles, H (alto) y L (bajo). Asigna valores a cada vocal tónica, además de las fronteras de las frases. En la Tabla 2 se indica la asignación simbólica. Se distinguen las siguientes situaciones: pausa inicial de frase con asignación ‘H[‘; una frontera de frase menor correspondiente a los signos de puntuación “:” y “-“, con asignación ‘L-H]’; final de frase correspondiente al signo de puntuación “.”, asignada por ‘L-L]’, y vocal acentuada, indicada por ‘H*’.

En la Tabla 3 se indica la asignación numérica, adaptado de valores dados en Rodríguez et al (1984).

Tabla 3. Conversión de valores simbólicos de entonación en valores numéricos

Condición o valor simbólico de entonación	Valor numérico de entonación (%,f0)
Pausa inicial	(0,120)
Pausa final	(99,80)
Vocal tónica, H*<0.5>	Se asigna (30, 130) a la vocal, y (0,100) a la consonante inicial de la misma sílaba; si esta consonante no existe, entonces se asigna también a la vocal tónica
L-H]<d.d>	Se asigna (90,100) al fonema anterior a la puntuación
L-L]<d.d>	Se asigna (99,80) al fonema anterior a la puntuación
H*<0.5> L-H]<d.d>	Se asigna (30, 130) (80, 120) a la vocal, (0,100) a la consonante inicial de la misma sílaba; si esta consonante no existe, entonces se asigna también a la vocal tónica, y (99,100) al fonema anterior a la puntuación
H*<0.5> L-L]<d.d>	Se asigna (30, 130) (80, 90) a la vocal, (0,100) a la consonante inicial de la misma sílaba; si esta consonante no existe, entonces se asigna también a la vocal tónica, y (99, 80) al fonema anterior a la puntuación

El efecto combinado de aplicar estas dos asignaciones a una frase declarativa típica se puede resaltar así: la entonación parte de una pausa inicial H[con un valor de 100 Hz. En cada vocal tónica se llega a un pico de f0 (H*) de 130 Hz a un 30 % del segmento. A la sílaba tónica de una vocal tónica se le asigna un f0 de 100 Hz en su inicio, bien sea si se inicia con una consonante, o la misma vocal tónica. Si se presenta una puntuación que marca una pausa, como una coma o paréntesis, se le asigna un f0 de 100 al 90% del segmento anterior a la puntuación. En cambio, si se presenta una puntuación final de frase, tipo punto “.” o punto y coma “;”, entonces se asigna un valor de f0 = 80 Hz al final del fonema anterior a la puntuación. De momento no se distingue frase declarativa de interrogativa.

2.4 Otros aspectos de la síntesis

En la ejecución de nuestro programa, después de realizar los pasos anteriores, se procede con las siguientes tareas: 1) asignación de duración, 2) cambios fonéticos por contexto, 3) transformación a la representación SAMPA.

Las duraciones de los fonemas para nuestro sistema, obtenidos de Mora (1996), se indican en la Tabla 4. Es de hacer notar que todos los fonemas tienen un solo valor de duración, con excepción de las vocales, para las cuales hay mayor duración en caso de ser acentuadas, fenóme no bien conocido, reportado por Mora (1998).

Tabla 4. Duraciones de los fonemas.

Fonema	Duración en ms	Fonema	Duración en ms
a	90	B	65
e	90	f	100
i	80	t	85
o	90	d	60
u	80	D	65
a:	108	S	100
e:	108	k	100
i:	96	g	50
o:	108	G	80
u:	96	x	130
j	60	m	70
W	60	n	80
w	45	~	110
y	90	N	50
c	135	M	50
s	110	l	80
s2	110	L	105
z	60	r	50
p	100	R	80
b	60		

El habla es continua, por lo tanto en la transcripción hay que tener en cuenta los efectos de unión entre palabras (coarticulación, elisiones, asimilaciones). A continuación se indican los cambios fonéticos que se realizan hacia el final del programa en función de contexto (Obediente, 1998).

- la “d”, “g”, “b” sonora, fricativa se reemplaza respectivamente por la “d”, “g”, “b” sonora, oclusiva, cuando viene después de pausa, “m” o “n”.
- en caso de dos consonantes seguidas, se elimina una (homologación).
- en caso de fonema “k”en distensión se reemplaza por “g”.
- en caso de fonema “k” seguido por “s”, siendo la “s” el último fonema de sílaba, la “k” se reemplaza por “g”.
- en caso de fonema “p”en distensión se reemplaza por “b”.
- en caso de fonema “p” seguido por “s”, siendo la “s” el último fonema de sílaba, la “p” se reemplaza por “b”.
- en caso de fonema “t”en distensión se reemplaza por “d”.
- en caso de fonema “t” seguido por “s”, siendo la “s” el último fonema de sílaba, la “t” se reemplaza por “d”.
- la “i” en contacto con vocal acentuada o “a”, “e”, u “o”, se reemplaza por la glide “j”.
- la “u” en contacto con vocal acentuada o “a”, “e”, u “o”, se reemplaza por la glide “w”.
- “s” o “z” en distensión se reemplaza por “s2”.

Después de haber manejado internamente las representaciones de los fonemas, al final se convierten a su representación SAMPA, que se describió anteriormente.

2.5 Descripción de la estructura del programa

En la Fig. 2 se muestra un diagrama de flujo de información entre las subrutinas que forman parte del programa.

En la Tabla 5 se describe la función de cada una de las subrutinas, como también se indica la variable que sirve de entrada y de salida para cada una de estas subrutinas. Se hace notar que el programa es capaz de manejar un texto de cualquier tamaño, con cualquier cantidad de frases. En la subrutina “Process_and_Send_to_Synth” se desprenden las frases, una por una, se mandan a “Process_Sentence” donde se realiza todo el procesado de la frase, el resultado es devuelto a “Process_and_Send_to_Synth”, que a su vez lo despacha a “Use_Synth”, y así se continua hasta que el texto completo se procesa y se finaliza el programa, con la generación del archivo .pho.

En la Tabla 6 están los resultados internos de las variables importantes con la frase: “Este atlas no es étnico.”. Se explica la Tabla 6 haciendo referencia a la Tabla 5. Entonces la primera columna a la izquierda, titulada @tokens, ha sido desprendida del texto de entrada por la subrutina *Process_and_Send_to_Synth* y por la subrutina *Process_Sentence* y es la entrada de la subrutina *treat*. Esta subrutina simplemente cambia mayúsculas a minúsculas y genera el vector @tokens0. Este vector es procesado por *add_tags* que distingue palabras sin acentos (FW), con acentos (CW) y signos de puntuación (PUNCT) y genera @tokens1. Este a su vez es procesado por *phrases* que tiene la tarea de distinguir fronteras menores internas. En este ejemplo, no existe tal frontera menor por lo que @tokens2, la salida de esta subrutina, es igual a la entrada. A continuación el vector es procesado por *transcribe* , que a su vez es apoyada por varias subrutinas. El resultado de aquí es @tokens3 donde es posible apreciar varios eventos; incluyendo la división en sílabas, separadas por un punto, indicando igualmente las vocales con acentos, agregando dos puntitos “:” a la vocal. A continuación la subrutina *conv* convierte el vector @tokens3 en @tokens4: se separan los componentes del vector en fonos individuales se enumeran los fonos, se identifica cada fono con el número de sílaba, y se colocan valores simbólicos de entonación en las fronteras de la frase (principio y fin), y a cada vocal tónica. En la si

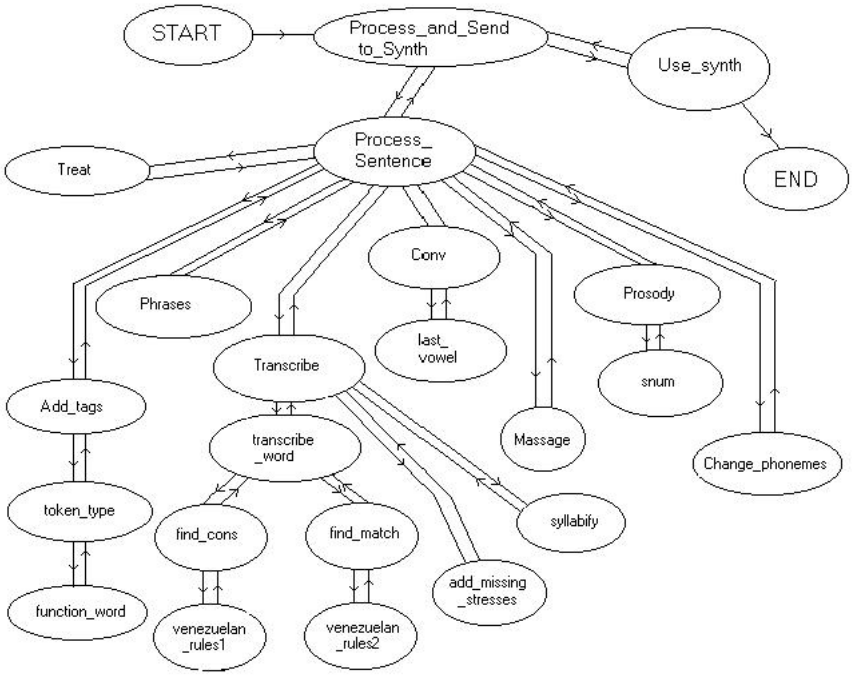

Fig. 2. Diagrama de Flujo de Intercambio de Información

Tabla 5. Subrutinas de TAFV.

Nombre	Descripción	Entrada/Salida
START	Se admite el texto de entrada	texto
process_and_send _to_synth	Se separa el texto en frases	texto/frase
process_sentence	Se reparte las tareas para el procesamiento de las frases	@tokens/@tokens7
treat	Se convierten todas las letras en minúsculas y se hace el preprocesado del texto descrito en la sección 2.1	@tokens/@tokens0
add_tags	Se distinguen palabras sin acentos (FW), con acentos (CW) y signos de puntuación (PUNCT)	@tokens0/@tokens1
token_type	Subrutina auxiliar de add_tags	
function_word	Subrutina auxiliar de token_type	
phrases	Se distinguen minor-phrase boundary (secuencia de CW-FW) para introducir un PUNCT sin duración	@tokens1/@tokens2
transcribe	Se reparte las tareas para la transcripción fonética de una frase	@tokens2/@tokens3
transcribe_word	Subrutina de transcribe: por medio de dos pasos se transcribe palabra por palabra el texto	
find_cons	Subrutina auxiliar de transcribe_word para transcripción fonética de ciertas consonantes: “c”, “g”, “y”, y elimina “u” muda.	
venezuelan_rules1	Archivo auxiliar de find_cons	
find_match	Subrutina auxiliar de transcribe_word para transcripción fonética inicial de	
venezuelan_rules2	Archivo auxiliar de find_match	
add_missing _stresses	Subrutina de transcribe: se coloca acento prosódico sobre las vocales tónicas que no llevan acento ortográfico	
syllabify	Subrutina de transcribe: coloca separador entre las sílabas	
conv	Coloca duración a fonemas y valor simbólico de entonación	@tokens3/@tokens4
last_vowel	Subrutina auxiliar de conv que detecta última vocal de la frase	
massage	Realiza cambios fonéticos por contexto	@tokens4/@tokens5
prosody	Reemplaza entonación simbólica por entonación numérica	@tokens5/@tokens6
snum	Subrutina auxiliar de prosody que determina el número de sílaba de la frase	
change_phonemes	Convierte representación fonética interna del programa por representación SAMPA	@tokens6/@tokens7
use_synth	Escribe resultados a archivo de salida .pho; en caso de haber otra frase por procesar, regresa a process_and_send_to_synth	@tokens7/test.pho

Tabla 6, Resultados internos de los variables importantes con la frase: "Este atlas no es étnico."

@tokens	@tokens0	@tokens1	@tokens2	@tokens3
0 'Este'	0 'este'	0 'este/CW'	0 'este/CW'	0 'e: s . t e/CW'
1 'atlas'	1 'atlas'	1 'atlas/CW'	1 'atlas/CW'	1 'a: . t l a s/CW'
2 'no'	2 'n\'o'	2 'n\'o/CW'	2 'n\'o/CW'	2 'n o:/CW'
3 'es'	3 'es'	3 'es/CW'	3 'es/CW'	3 'e: s/CW'
4 '\'etnico'	4 '\'etnico'	4 '\'etnico/CW'	4 '\'etnico/CW'	4 'e: t . n i . k o/CW'
5 '.'	5 '.'	5 './PUNCT'	5 './PUNCT'	5 './PUNCT'

@tokens4	@tokens5	@tokens6
0 '# 50 H⌈<1.0>'	0 '# 50 H⌈<1.0>'	0 "#°50°(0,120)"
1 'e: 0 H*<0.5>'	1 'e: 0 H*<0.5>'	"e:°108°(0,100)°(30,130)"
2 's 0 '	2 's2 0 '	2 "s2°110"
3 't 1 '	3 't 1 '	3 "t°85"
4 'e 1 '	4 'e 1 '	4 "e°90"
5 'a: 2 H*<0.5>'	5 'a: 2 H*<0.5>'	"a:°108°(0,100)°(30,130)"
6 't 3 '	6 't 3 '	6 "t°85"
7 'l 3 '	7 'l 3 '	7 "l°80"
8 'a 3 '	8 'a 3 '	8 "a°90"
9 's 3 '	9 's2 3 '	9 "s2°110"
10 'n 4 '	10 'n 4 '	10 "n°80°(0,100)"
11 'o: 4 H*<0.5>'	11 'o: 4 H*<0.5>'	11 "o:°108°(30,130)"
12 'e: 5 H*<0.5>'	12 'e: 5 H*<0.5>'	12 "e:°108°(0,100)°(30,130)"
13 's 5 '	13 's2 5 '	13 "s2°110"
14 'e: 6 H*<0.5>'	14 'e: 6 H*<0.5>'	14 "e:°108°(0,100)°(30,130)"
15 't 6 '	15 'd 6 '	15 "d°60"
16 'n 7 '	16 'n 7 '	16 "n°80"
17 'i 7 '	17 'i 7 '	17 "i°80"
18 'k 8 '	18 'k 8 '	18 "k°100"
19 'o 8 L-L⌋<1.0>'	19 'o 8 L-L⌋<1.0>'	19 "o°90°(99,80)"
20 '# 250'	20 '# 250 '	20 "#°250°(99,80)"

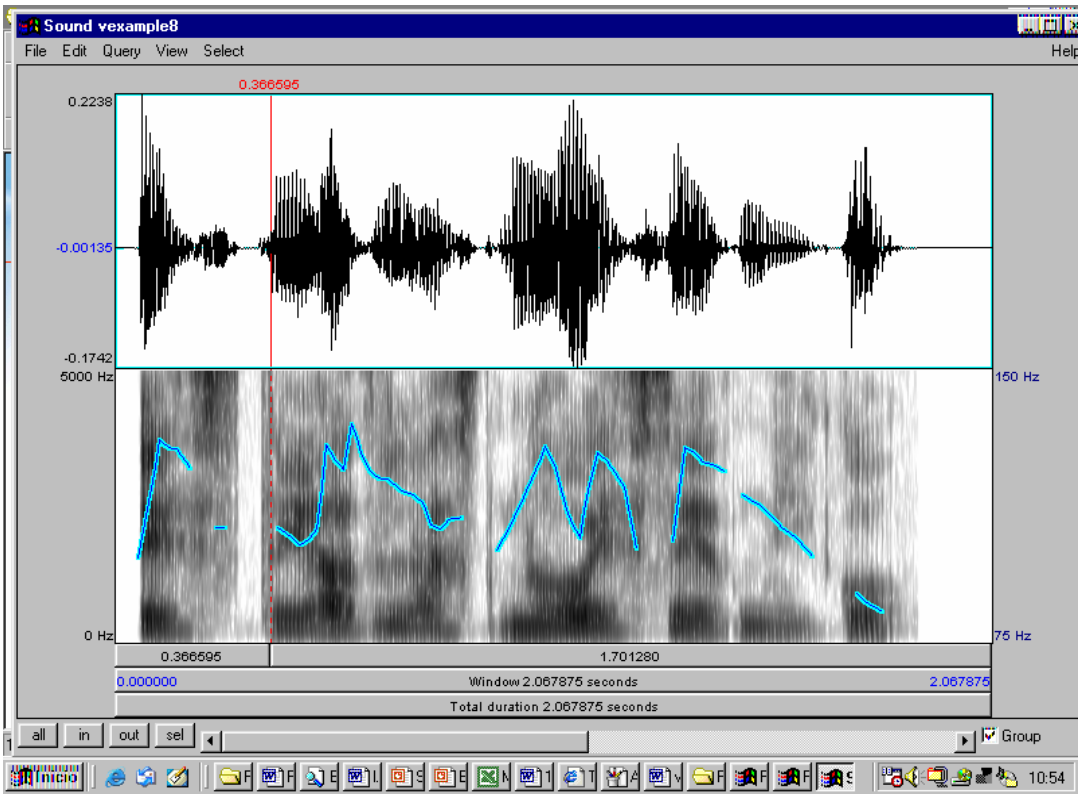

Fig. 3. Oscilograma de voz sintetizada de la frase "Este atlas no es étnico.", y abajo, espectrograma y función de f0.

guiente etapa, con la subrutina *massage*, se hace el procesado de cambios fonéticos por contexto, dando lugar al vector @tokens5. En la última columna de la Tabla 6 se indica el vector @tokens6, producto de la subrutina *prosody*. En este punto, ya se ha hecho el reemplazo de los valores simbólicos de la entonación por valores numéricos, y se ha agregado el valor de la duración de cada fono. Una última transformación se realiza con la subrutina *change_phonemes* que simplemente sustituye los fonos por su representación SAMPA, y lo descarga en @tonos7.
En la Figura 3 se muestra la forma de onda de la frase sintetizada "Este atlas no es étnico.", y abajo el espectrograma y la función de f0 correspondiente.

3 Conclusiones y recomendaciones

Se dispone de un programa que permite realizar la conversión texto a voz en forma automática en el dialecto venezolano del español. De momento ha permitido realizar gran cantidad de pruebas para establecer la calidad de los difonos de la base de datos vz1. Es un programa que admite una gran variedad de signos y caracteres especiales, además de texto normal, de modo que es capaz de leer la gran mayoría de páginas web en español sin mayor dificultad.

De esta forma, puede ser utilizado en una gran variedad de sistemas automáticos de respuesta oral. Una aplicación particularmente atractiva es como parte de un sistema de atención automática al usuario, es decir, combinándolo con un sistema de reconocimiento automático de voz y un sistema de inteligencia artificial, para poder atender automáticamente a un ser humano, y así cumplir con un servicio público.
También es de utilidad para personas discapacitadas: ciegos y sordos para dirigirse a oyentes no conocedores de la lengua de señas. A mediano plazo, se espera que pueda ser parte integral de una nueva línea de investigación de la prosodia del español hablado en Venezuela, permitiendo sintetizar frases con diferentes funciones de entonación, duración y ritmo para realizar pruebas de percepción.

4 Agradecimientos

Se agradece el financiamiento de este proyecto por parte de la agencia francesa ECOS-NORD (Action V99H01) y los institutos venezolanos CONICIT, CDCHT-ULA y Fundayacucho. También se agradece la ayuda del Profesor Andrés Arcia de la Universidad de Los Andes en resolver problemas de programación en Perl.

Referencias

Cavé C, Rodríguez M, Mora E, Clairet S y Hirst D, 2005, Un sistema de síntesis de habla en español de Venezuela, Proceedings of IX Simposio Internacional de Comunicación

Social, pp. 513-514.
Dutoit T, 1997, An introduction to Text to speech synthesis, Dordrecht, Klewer.
Huang X, Acero A y Hon H, 2001, Spoken language processing: A guide to theory, algorithm and system development, Pearson Edition.
Mora E, 1996, Caractérisation prosodique de la variation dialectale de l'espagnol parlé au Vénézuéla, Thèse de doctorat de l'Université de Provence.