Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Construcción y evaluación del back-end de un sistema de síntesis de habla en español argentino

Tesis de Licenciatura

Luisina Violante

Director: Dr. Agustín Gravano

Buenos Aires, Agosto de 2012

RESUMEN

Esta tesis consiste en la creación y evaluación de un back-end de síntesis de habla para el idioma español argentino. Los sistemas resultantes fueron creados con dos técnicas de síntesis distintas: síntesis por concatenación de unidades y síntesis basada en modelos ocultos de Markov. Se trabajó sobre un inventario de sonidos provisto por el Laboratorio de Investigaciones Sensoriales, INIGEM, CONICET-UBA, sobre el cual se realizaron modificaciones para mejorar la calidad de la voz sintetizada. Los sistemas fueron evaluados utilizando tests de inteligibilidad y naturalidad, contrastando los resultados con los obtenidos en trabajos previos. Finalmente, se utilizaron técnicas de procesamiento de señales para lograr habla expresiva y variaciones en la prosodia del habla sintetizada.

Palabras claves: Procesamiento del habla, síntesis del habla, síntesis concatenativa, síntesis basada en modelos ocultos de Markov

Índice general

Índice general III

Índice de tablas VI

Índice de figuras VII

1. Introducción 1

1.1. Sistemas de Texto-a-Voz 1

1.2. Tipos de síntesis 3

1.2.1. Síntesis concatenativa 3

1.2.1.1. Tipos de unidades 3

1.2.2. Síntesis basada en modelos ocultos de Markov 4

1.2.3. Otros tipos de síntesis 5

1.3. Antecedentes 5

1.3.1. Actualidad 8

1.4. Objetivo del trabajo 8

1.5. Descripción general del trabajo realizado 8

2. Sistemas considerados 10

2.1. Festival TTS 10

2.2. MARY TTS 12

2.3. FreeTTS 13

3. Corpus SECYT **14**

3.1. Alfabeto fonético 14

3.1.1. Inventario fonético utilizado en la creación de voces . . 16

4. Desarrollo del trabajo **18**

4.1. Modificaciones realizadas al cuerpo de datos SECYT 18

4.1.1. Etiquetados 18

4.1.2. Cobertura de difonos 19

4.1.3. Acentos léxicos 19

4.1.4. Grabaciones 20

4.1.5. Otras modificaciones menores 24

4.2. Creación de voces con Festival TTS 24

4.2.1. Clunits 25

4.2.2. CLUSTERGEN - Síntesis estadística paramétrica . . . 28

4.3. Creación de voces con MARY TTS 29

4.3.1. Difonos 29

4.3.2. HMM 31

5. Evaluación de los sistemas **33**

5.1. Tests y diseño experimental 33

5.2. Test MOS (*Mean Opinion Score*) 34

5.3. Test SUS (*Syntactically Unexpected Sentences*) 35

5.4. Resultados 35

5.4.1. Test de naturalidad MOS 36

5.4.2. Test de inteligibilidad SUS 39

5.5. Resumen de los resultados 42

5.6. Comparación con los resultados de Gurlekian et al. 2012 . . . 43

6. Habla expresiva **44**

6.1. Lenguajes de marcado 44

6.2. Procesamiento de señales para generar habla expresiva 45

7. Conclusiones **48**

7.1. Balance del trabajo 48

7.2. Trabajo futuro 49

Anexos **50**

Referencias **61**

Índice de tablas

3.1. Alfabeto SAMPA para el español de la Argentina 15
3.2. Inventario fonético utilizado para la creación de voces 17

4.1. Tabla de conversión de tono 22

5.1. Sistemas construidos 34
5.2. Resultados del "test choice" 38
5.3. Resultados de inteligibilidad SUS para cada voz 40
5.4. Palabras erradas en evaluación SUS, sintetizadas con la voz HMM_ORIG 41

6.1. Características acústico-prosódicas para habla expresiva . . . 46

Índice de figuras

1.1. Espectrogramas y formas de onda de fonos y difonos 4
1.2. Máquina de habla acústico-mecánica 6
1.3. The Voder 7

4.1. Funciones para reducir picos tonales 23
4.2. Ejemplo - Reducción de picos tonales 23

5.1. Evaluaciones MOS para cada voz 37
5.2. Evaluaciones MOS para cada voz 37
5.3. Media de puntajes SUS para cada sistema 40

AGRADECIMIENTOS

A Agustín Gravano. Sin duda el mejor director que podría haber elegido. Su entusiasmo contagioso, buena voluntad y sus consejos, lograron transformar "la temida tesis" en una experiencia muy gratificante.

A los profesores de esta carrera maravillosa, quienes ponen todo su empeño y predisposición para día a día formarnos como futuros profesionales.

A todos los amigos que me dio esta facultad; ya que sin ellos, todo el proceso hubiese sido más difícil y más aburrido. Especialmente le quiero agradecer a Facu por ser un gran amigo y por estar siempre que lo necesité.

A mis amigos Seba, Karin y Mau; con los cuales compartí y disfruté los inicios de mi carrera universitaria. Y a Sandra y Juan, quienes sin saberlo, marcaron la metodología de estudio que considero me llevó al éxito durante estos años. Con ellos aprendí que un tropezón no es caída, y que con empeño y dedicación, todo es posible.

A mis amigas: Nati, Ro, Gi, Ani, Lu, Lis, Dai, Meli, Eli y Flor. Por los tantos momentos, charlas y noches de chicas, que me llenan de alegría.

A toda mi familia. Por estar siempre conmigo, apoyándome incondicionalmente en cada paso que doy.

A Germán, por confiar en mi y ser una persona maravillosa que ilumina mi vida.

Por último, y no menos importante, a Reichi, a mamá y a papá. Simplemente por ser como son, por amarme y acompañarme siempre. Los adoro.

Para Germi

Capítulo 1

Introducción

1.1. Sistemas de Texto-a-Voz

Un SISTEMA TTS (del inglés *text-to-speech*) es un sistema que convierte texto de entrada en habla. Un sistema de estas características consta de dos partes principales: un *front-end* y un *back-end*.

El ***front-end*** se encarga de procesar el texto de entrada con el objetivo de reunir información necesaria para generar la secuencia de fonemas con anotaciones que posteriormente se utilizará para la síntesis. Los FONEMAS son clases abstractas de sonidos que permiten distinguir palabras de un idioma. Por ejemplo, la palabra "hueco" se pronuncia /weko/ y se diferencia de /weso/ en un solo fonema.

Inicialmente se normaliza el texto. Para ello es necesario identificar y segmentar las oraciones teniendo en cuenta los puntos de fin de oración u otros símbolos, como por ejemplo, los signos de interrogación o exclamación. Luego es necesario tokenizar las palabras y expandir las abreviaturas, siglas, acrónimos[1] y expresiones numéricas. Posteriormente, se realiza un análisis lingüístico para identificar la estructura sintáctica (sujeto y predicado) y clases de palabras (sustantivo, verbo, adjetivo, preposición, adverbio, artículo, interjección, pronombre, conjunción). El resultado es texto anotado. El siguiente paso consiste en realizar un análisis fonético sobre ese texto, convirtiendo grafemas (las unidades mínimas e indivisibles de la escritura de una

[1] Acrónimo: Palabra formada por las letras iniciales de una expresión compuesta, pero que suele ajustarse a las reglas fonológicas de la lengua.

lengua) a fonemas. Para la conversión se pueden utilizar reglas letra-a-sonido (LTS: *letter-to-sound*) o diccionarios de palabra-transcripción. Además, debe desambiguar homógrafos y tiene que estar contemplada la posibilidad de convertir palabras extranjeras o palabras fuera de vocabulario (OOV: *out of vocabulary*). Como resultado de este paso se obtienen fonemas anotados. Por último se realiza un análisis para asignar: frases prosódicas utilizando puntuación y sintaxis; acentos prosódicos según fueran palabras de contenido (sustantivos, verbos, adjetivos y la mayoría de los adverbios) o de función (preposiciones, pronombres, verbos auxiliares, conjunciones, artículos gramaticales, entre otros); y contorno entonacional (declarativo, interrogativo, etc). El resultado final es una secuencia de fonemas anotados con información de control y características acústicas y prosódicas relevantes, tales como:

- la FRECUENCIA FUNDAMENTAL (F0), que se define como la frecuencia más baja de una onda periódica, de modo tal que las frecuencias dominantes pueden expresarse como múltiplos de la misma, y tiene una correlación fuerte con la percepción del nivel tonal de la voz;
- la INTENSIDAD, que se define como la potencia acústica transferida por una onda sonora por unidad de área normal a la dirección de propagación, y tiene una correlación fuerte con la percepción del volumen de la voz;
- la DURACIÓN de cada segmento del habla (sílaba, fonema, etc.);
- la CALIDAD DE LA VOZ, es decir las características de la fuente emisora del sonido, que tiene relación con el timbre vocal.

Por otro lado, el ***back-end*** es el módulo encargado de la síntesis del habla propiamente dicha. Toma como input la secuencia de fonemas con anotación prosódica resultante del *front-end* y genera un archivo de audio con el texto sintetizado.

Para la síntesis, existen varios métodos, muy distintos entre sí. Cada uno posee ventajas y desventajas, y su elección depende fuertemente de la calidad de los resultados esperados, del material disponible para su implementación, y del poder de cómputo y el espacio de almacenamiento disponibles.

1.2. Tipos de síntesis

A continuación, se dará una breve descripción de los diferentes tipos de síntesis de habla para introducir al lector en el tema.

1.2.1. Síntesis concatenativa

La síntesis concatenativa se basa, como indica su nombre, en la concatenación de unidades de habla previamente grabadas En primer lugar, se graban oraciones preparadas y se recortan las unidades, las cuales pueden ser frases, palabras, sílabas, fonemas, etc. Luego, al querer sintetizar un texto, se seleccionan unidades dentro del inventario y se concatenan.

Los sistemas de este tipo seleccionan para la síntesis aquellas instancias de la base de datos de unidades que posean las características prosódicas más parecidas a las deseadas; es decir, se busca maximizar una función objetivo. Luego, tener en el inventario un mapeo completo de todas las unidades existentes en el lenguaje y varias instancias de cada unidad, resulta relevante para la calidad del habla sintetizada. Es por ello que la buena elección de las oraciones a grabar es una tarea fundamental.

La síntesis concatenativa es uno de los métodos más comunes hoy en día y el que produce los resultados más naturales. Resulta relativamente barato y fácil de implementar en comparación con otros paradigmas de síntesis, aunque se tiene escaso control sobre la variabilidad prosódica, y el habla sintetizada se asemeja al estilo del habla utilizado en las grabaciones de la base de datos. A medida que se requiera más variación en el estilo y la emoción del habla, para conservar la calidad de la selección de unidades, será necesario grabar bases de datos más y más grandes con diferentes estilos a fin de lograr la síntesis deseada. Por otro lado, cambiar de idioma implicaría, entre otras cosas, regrabar todo el inventario.

1.2.1.1. Tipos de unidades

Como se mencionó anteriormente, las unidades a concatenar podrían ser palabras, frases, sílabas, fonemas o difonos, entre otros.

Los FONOS son los sonidos básicos de un idioma, por ejemplo “dedo” se pronuncia [deδo]. En cambio, los fonemas son clases abstractas de palabras que permiten distinguir palabras de un idioma. Por ejemplo, la palabra “de-

do" consta de tres fonemas (/dedo/: /d/, /e/, /o/). En esta palabra, el fonema /d/ tiene dos instanciaciones (llamadas ALÓFONOS): [d] en la primera sílaba, y [δ] en la segunda. Un DIFONO es aquel sonido que queda delimitado por aproximadamente la mitad de un fono hasta aproximadamente la mitad del siguiente. En la figura 1.1 se muestra, a modo de ejemplo, el espectrograma de la palabra "mamá" junto con el etiquetado de fonos y difonos.

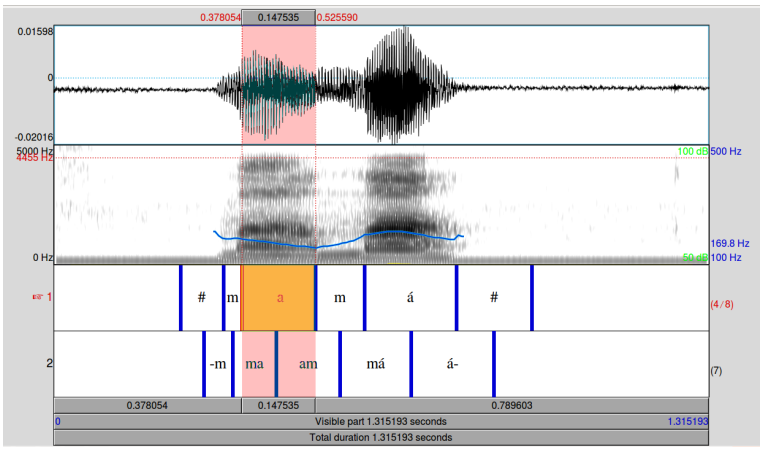

Figura 1.1: *Ejemplo - Espectrograma y forma de onda de la palabra "mamá", junto con su respectivo etiquetado de fonos y difonos.*

1.2.2. Síntesis basada en modelos ocultos de Markov

Los modelos ocultos de Markov (HMM, del inglés *Hidden Markov Models*) son empleados para modelar y generar información sobre el espectro, la frecuencia fundamental, y la duración de los segmentos del habla[2]. La información espectral es suficiente para generar la forma de onda correspondiente. Es decir, dada una secuencia de fonos, las secuencias de parámetros de habla son generadas a partir de HMM directamente basados en el criterio de máxima verosimilitud, es decir que se elegirán aquellos valores que tengan mayor probabilidad de ocurrencia según los datos observados. Al considerar la relación entre los parámetros estáticos y dinámicos, se generan secuencias espectrales suaves (ver figura 1.1) de acuerdo a las estadísticas de esos parámetros modelados por los HMMs. Como resultado, se produce el sonido resultante de los patrones espectrales de la secuencia de fonos de entrada.

[2]Para una descripción detallada de HMM ver [15] y el capítulo 9 de [1].

Una ventaja que presenta el modelo es que requiere poco espacio de almacenamiento comparado con la síntesis concatenativa, y además permite modelar diferentes estilos de habla sin necesidad de agrandar la base de datos.

1.2.3. Otros tipos de síntesis

La SÍNTESIS ARTICULATORIA se basa en modelos computacionales del tracto vocal, requiere modelos mecánicos y acústicos de la producción del habla: vibración de las cuerdas vocales, aspiración del aire, movimiento de los articuladores (lengua, labios, etc) [11] [17]. Produce resultados inteligibles y con amplio control, pero todavía lejos de sonar naturales. Estos cómputos resultan muy costosos computacionalmente. Sin embargo, tienen la ventaja de ser livianos (ocupan poco espacio) y de presentar bajo costo en caso de querer cambiar de idioma.

La SÍNTESIS DE FORMANTES, también conocida como síntesis paramétrica o acústica, genera habla artificial aplicando una serie de filtros a una fuente de sonido (se modifican parámetros como ser la frecuencia fundamental y los niveles de ruido) [4] [3]. Los sistemas de esta característica comparten las mismas ventajas y desventajas que aquellos basados en síntesis articulatoria.

1.3. Antecedentes

La idea de que una máquina genere habla ha estado con nosotros desde hace bastante tiempo. Los intentos más antiguos de los que se tiene registro datan de fines del siglo XVIII.

En 1779, el científico danés Christian Gottlieb Kratzenstein, construyó modelos del tracto vocal que podrían producir las cinco vocales largas (a, e, i, o, u). Más tarde, en 1791, Wolfgang von Kempelen describió en su obra *Mechanismus der menschlichen Sprache nebst der Beschreibung seiner sprechenden Maschine* (Mecanismo del lenguaje humano, junto con una descripción de una máquina parlante) una máquina accionada con un fuelle. Su trabajo agregaba, además de modelos del tracto vocal, modelos de la lengua y los labios, para producir tanto vocales como consonantes.

En 1837, Charles Wheatstone creó una máquina parlante acústico-mecánica basada en el diseño de von Kempelen (Figura 1.2). Esta máquina, bien manejada, podía producir palabras y hasta frases enteras. A partir de entonces,

se construyeron otras máquinas que también modelaban el tracto vocal humano, lengua, labios y faringe.

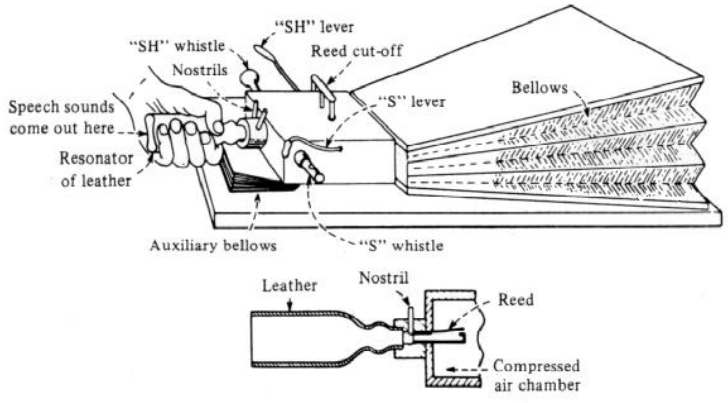

Figura 1.2: *Máquina de habla acústico-mecánica*[3]

En los años 30, en los laboratorios Bell Labs se desarrolló el VOCODER (*Voice Coder*), un analizador y sintetizador del habla operado por teclado que era claramente inteligible. En 1939, Homer Dudley refinó este dispositivo y creó *VODER* (*Voice Operating Demonstrator*), exhibido en la Feria Mundial de Nueva York de 1939 (Figura 1.3). Este fue el primer dispositivo en generar habla humana electrónicamente, y el primero en ser considerado un sintetizador de habla. El Voder constaba de una barra de muñeca para seleccionar una fuente de sonido o ruido y un pedal para controlar la frecuencia fundamental (F0). La señal pasaba a través de diez filtros de banda, cuyos niveles de salida eran controlados por los dedos. Requería mucho entrenamiento del operador poder reproducir una frase. La calidad y la inteligibilidad del habla estaban lejos de ser buenas, pero era evidente el potencial para producir habla artificial.

[3] http://www.haskins.yale.edu/featured/heads/SIMULACRA/kempelen.html

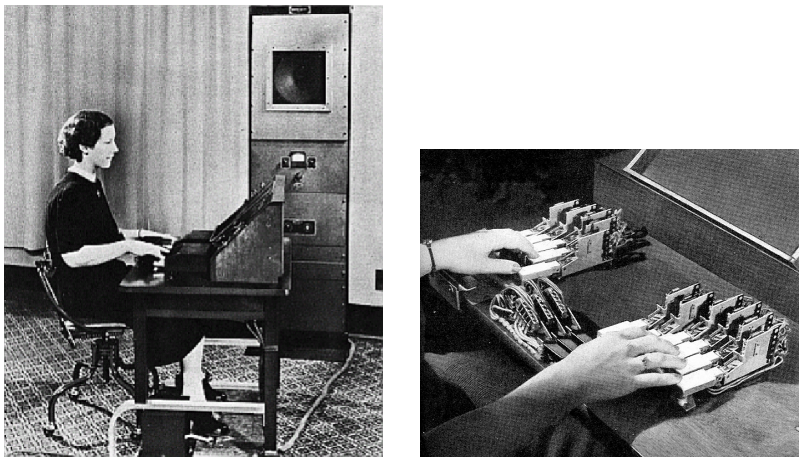

Figura 1.3: *The Voder*[4]

En 1950, Franklin S. Cooper terminó de construir *Pattern Playback* en los laboratorios Haskins. El objetivo era estudiar el efecto perceptual de detalles espectrales. Esta máquina convertía imágenes de patrones acústicos del habla en forma de espectrograma, en sonido. A partir del uso de este dispositivo, Alvin Liberman, Cooper Frank, Pierre Delattre y otros, fueron capaces de descubrir indicadores acústicos para la percepción de segmentos fonéticos (consonantes y vocales). Esta investigación fue fundamental para el desarrollo de técnicas modernas de síntesis de voz, máquinas de lectura para ciegos, para el estudio de percepción del habla y reconocimiento de voz, y para el desarrollo de la teoría motora de la percepción del habla[5].

El primer sintetizador articulatorio fue creado en 1958 por George Rosen en el Instituto de Tecnología de Massachusetts (M.I.T.). *DAVO* (*Dynamic Analog of the Vocal tract*) constaba de circuitos analógicos para modelar diferentes partes del tracto vocal, y era controlado por grabaciones de señales de control creadas a mano. Algunos años después, Dennis Klatt adaptó DAVO para ser controlado por computadora.

El primer sistema texto-a-habla completo para el idioma inglés, fue desarrollado en Japón en 1968 por Noriko Umeda. El mismo se basaba en modelos articulatorios e incluía módulos de análisis sintácticos con heurísticas sofisticadas. El habla resultante era bastante inteligible pero monótona y estaba lejos de tener la calidad de sistemas actuales.

[4] http://www.davidszondy.com/future/robot/voder.htm

1.3.1. Actualidad

Hoy en día existen varios sistemas TTS comerciales que soportan el idioma español, como ser AT&T[5] y Loquendo[6]. Sin embargo, son pocos los TTS gratuitos y sin fines de lucro que permiten la lectura, modificación y extensión del código. Un ejemplo de estos últimos es Festival[7], que cuenta con una voz masculina en español ibérico.

1.4. Objetivo del trabajo

El principal objetivo de este trabajo es construir el *back-end* de un sistema TTS con una voz en español argentino que pueda utilizarse a futuro como base para experimentar en las distintas áreas de conocimiento.

Para ello, se trabajará sobre sistemas de código abierto (*open source*), adaptándolos para que incorporen una voz argentina. Como todo sistema de síntesis, se busca obtener inteligibilidad y naturalidad en el habla sintetizada.

1.5. Descripción general del trabajo realizado

En el transcurso del trabajo, se crearon cinco voces basadas en concatenación de unidades, y tres voces basadas en síntesis paramétrica. En total, seis de ellas fueron construidas a partir de Festival TTS, y dos utilizando MARY TTS. Festival y MARY TTS son sistemas gratuitos y de código abierto, que ofrecen plataformas generales para la construcción de sistemas de síntesis de habla. Todos los sintetizadores creados, se construyeron a partir del cuerpo de datos SECYT, creado por el Laboratorio de Investigaciones Sensoriales, INIGEM, CONICET-UBA [18].

Durante el desarrollo del trabajo, se enmendaron errores encontrados en la base de datos y se efectuaron modificaciones en los audios, con el propósito de obtener grabaciones más adecuadas para utilizar en la creación de sistemas TTS. Luego, se evaluaron dichos sistemas utilizando tests de inteligibilidad y de naturalidad, para medir la calidad general.

[5] http://www2.research.att.com/~ttsweb/tts/
[6] http://www.loquendo.com
[7] http://www.cstr.ed.ac.uk/projects/festival/

Finalmente, se describen los intentos por obtener habla emotiva y variaciones prosódicas en el habla sintetizada, para lograr un habla que se asemeje más al habla natural. Para ello, se utilizaron lenguajes de marcado sobre el texto a sintetizar, para indicar qué porciones del texto sintetizar de qué manera; y se buscaron indicadores acústico-prosódicos que describieran las emociones o expresiones deseadas, para luego modificar los audios sintetizados a través de técnicas de procesamiento de señales.

En el Capítulo 2, se describen las plataformas tenidas en cuenta en el trabajo para la construcción de las voces. En el Capítulo 3, se describe el corpus utilizado en la creación de voces, así como también el alfabeto fonético empleado en las transcripciones fonéticas del mismo. En el Capítulo 4, se detalla el desarrollo de creación de voces dentro de cada plataforma, junto con los problemas encontrados durante el proceso y las soluciones implementadas en cada caso. En el Capítulo 5, se presentan los tests utilizados para la evaluación de los sistemas construidos y se exhiben los resultados obtenidos a partir de dicha evaluación. En el Capítulo 6, se exponen diferentes técnicas para obtener variabilidad prosódica en el habla sintetizada y lograr un habla expresiva. Por último, en el Capítulo 7, se presentan las conclusiones del trabajo realizado.

Capítulo 2

Sistemas considerados

Los sistemas considerados para el trabajo fueron seleccionados por ser de código abierto (*open source*); lo que nos permite ver, modificar y extender el código existente. Los mismos ofrecen una plataforma para la construcción de sistemas TTS de propósito general para múltiples lenguajes. A continuación explicaremos brevemente los sistemas utilizados.

2.1. Festival TTS

Festival fue desarrollado por el CSTR (*The Center of Speech Technology Research*) de la Universidad de Edimburgo. Ofrece una plataforma general para la construcción de sistemas TTS, así como también incluye ejemplos de varios módulos. Es multilingüe; actualmente soporta, entre otros, el idioma inglés (británico y americano) y el español, aunque el inglés es el más avanzado en cuanto a su desarrollo. Nuevos lenguajes pueden ser creados para el sistema, y las herramientas y documentación necesarias para la construcción de nuevas voces están disponibles a través del proyecto FestVox[1] de Carnegie Mellon.

El sistema está escrito en C++ y usa *Edinburgh Speech Tools Library*[2] para la arquitectura de bajo nivel. *Edinburgh Speech Tools Library* es una biblioteca para software general de habla, escrita en C++ y también desarrollada en el CSTR de la Universidad de Edimburgo. Provee una serie de

[1] http://festvox.org
[2] http://www.cstr.ed.ac.uk/projects/speech_tools/

herramientas para realizar tareas comunes encontradas en el procesamiento del habla y proporciona un conjunto de programas ejecutables en forma independiente (*stand alone*) y un conjunto de llamadas a bibliotecas que pueden ser vinculadas a los programas de usuario.

Para una fácil especificación de parámetros y control de flujo, Festival utiliza como intérprete de comandos un lenguaje llamado Scheme (SIOD). Scheme es un lenguaje funcional y un dialecto de Lisp, posee una sintaxis simple y reducida. La utilización de un lenguaje con las características de Scheme, permite que gran parte de las funcionalidades de Festival sean totalmente controlables en tiempo de ejecución sin tener que volver a recompilar el sistema.

Festival es software libre, y tanto Festival como Speech Tools son distribuidas bajo una licencia de tipo X11[3].

Festival permite la configuración externa de módulos que son dependientes del lenguaje:

- Inventario fonético (conjunto de fonemas).
- Léxico (listado de palabras).
- Reglas letra-a-sonido (cómo pronunciar las palabras).
- Tokenización (definición de qué es una palabra).
- Etiquetado de clase de palabra (cómo decidir si una palabra es un verbo, sustantivo, adjetivo, etc.).
- Entonación y duración (características prosódicas del lenguaje).

Además, Festival permite los siguientes tipos de síntesis, entre otros:

- Síntesis por concatenación de fonos simples.
- Síntesis por concatenación de difonos.
- Síntesis HMM.

[3] http://en.wikipedia.org/wiki/MIT_License

En este trabajo se utilizaron dos módulos de Festival llamados *Clunits unit selection engine* para armar sistemas de síntesis concatenativa y *Clustergen parametric synthesis engine* para armar sistemas basados en HMM. Además, se creó un nuevo lenguaje llamado "spanish_arg" definiendo los módulos correspondientes al mismo.

El proceso básico de síntesis en Festival se reduce a la ejecución secuencial de un conjunto de módulos sobre una estructura llamada *utterance*. Cada utterance tiene asociado un conjunto de ítems, características y relaciones que, luego de aplicar todos los módulos, permite generar la forma de onda deseada. Cada módulo tiene acceso a esa estructura y puede generar más relaciones e items para enriquecerla.

2.2. MARY TTS

MARY (*Modular Architecture for Research on Speech Synthesis*)[4] es una plataforma escrita en Java, multilingüe y de código abierto, para sistemas TTS. Originalmente fue desarrollada como un proyecto colaborativo entre el laboratorio de Tecnologías del Lenguaje del Centro Alemán de Investigación de Inteligencia Artificial (DFKI) y el Instituto de Fonética en la Universidad de Saarland; y ahora esta siendo mantenida por el DFKI.

Actualmente MARY TTS soporta los idiomas alemán, inglés británico y americano, telugú, turco y ruso; y nuevos idiomas están siendo preparados. MARY TTS brinda herramientas para agregar soporte para nuevos lenguajes de manera rápida, y para construir voces basadas en sistemas de selección de unidades y HMM.

MARY está compuesto por distintos módulos y tiene la capacidad de parsear lenguajes de marcado (*markup languages*) para síntesis de habla, como SABLE (ver Sección 6.1 para más información sobre lenguajes de marcado). Además, provee una interfaz web accesible desde cualquier lugar sin necesidad de tener una instalación local del sistema.

La plataforma permite, gracias a su arquitectura modular, un procesamiento paso a paso respecto a la aplicación de los módulos involucrados en el proceso de síntesis, con acceso a los resultados parciales. Esto lo logra utilizando una representación interna de los datos llamada MaryXML, basada en XML (*eXtensible Markup Language*), que le da flexibilidad al sistema

[4] http://mary.dfki.de/

y le permite al usuario la posibilidad de modificar los datos en cada paso.

A diferencia de Festival, MARY TTS cuenta con una GUI (*Graphical User Interface*) para la síntesis y otra GUI intuitiva y especialmente diseñada para facilitar la creación de voces a usuarios inexpertos en el área.

2.3. FreeTTS

FreeTTS es un sistema de síntesis de habla escrito enteramente en el lenguaje de programación Java. Está basado en Flite: un pequeño sistema TTS derivado de Festival, de la Universidad de Edimburgo, y del proyecto FestVox, de la Universidad Carnegie Mellon.

Incluye soporte para voces en inglés americano (una voz en inglés americano, masculina, 8khz difonos; una voz en inglés americano, masculina, 16khz difonos; una voz en inglés americano, masculina, 16khz dominio limitado), para voces MBROLA[5] (una voz femenina y dos masculinas en inglés americano de 16khz), y además permite importar voces desde FestVox.

Una vez obtenidos los sistemas en Festival, se quiso importar la voz a FreeTTS. Sin embargo no fue posible ya que sólo permite importar voces en inglés americano. En consecuencia, lamentablemente no pudimos construir una voz con el sistema FreeTTS.

[5] http://tcts.fpms.ac.be/synthesis/

Capítulo 3

Corpus SECYT

Para la creación de las voces se utilizó el cuerpo de datos SECYT, construido por el Laboratorio de Investigaciones Sensoriales (LIS), INIGEM, CONICET-UBA [18] [9]. El mismo se realizó en base a 741 oraciones declarativas diseñadas específicamente para realizar un estudio sobre la prosodia del habla. Cada oración tiene, en promedio, 7 palabras y una duración de 3,9 segundos.

El cuerpo de datos contiene tres tipos de archivos: ".wav", ".fon" y ".gra". Los archivos ".wav" corresponden a las grabaciones realizadas por una locutora argentina de las oraciones previamente seleccionadas. Cada audio tiene asociados dos archivos, un ".fon" y un ".gra", conteniendo las transcripciones fonética y de palabras respectivamente, alineadas temporalmente al audio.

El material obtenido tiene un valor inconmensurable dado el tiempo que lleva realizarlo y la compleja tarea que implica seleccionar las frases a grabar. Además, son muchas las ventajas de poseer datos etiquetados en forma manual frente al etiquetado automático, principalmente al momento de crear la base de datos de sonidos que será utilizada por el TTS al sintetizar el habla, logrando una mejor calidad vocal del mismo.

3.1. Alfabeto fonético

Para el etiquetado fonético de la base de datos SECYT, se utilizó de una adaptación al español del alfabeto fonético SAMPA (*Speech Assesment Methods: Phonetic Alphabet*) [9]. La tabla 3.1 indica, para cada fonema, la

descripción del modo (de qué forma se produce el sonido en el tracto vocal) y el punto (en qué región del tracto vocal se produce el sonido) de articulación, si es sonoro o sordo, y se exhibe una palabra a modo de ejemplo junto a su transcripción ortográfica.

		SAMPA para la Argentina		
N	**Sampa**	**Modo, Punto, F0**	**Transc.**	**Palabra**
1	**i**	Vocal, Cerrada-anterior, sonora	b**i**s	bis
2	**e**	Vocal, cerrada-media-anterior, sonora	m**e**s	mes
3	**a**	Vocal, abierta-central, sonora	m**a**s	más
4	**o**	Vocal, cerrada-media-posterior-redondeada, sonora	t**o**s	tos
5	**u**	Vocal, cerrada-posterior-redondeada, sonora	t**u**l	tul
6	**j**	Aproximante, palatal, sonora	laB**j**o	labio
7	**w**	Aproximante, labial velar, sonora	aG**w**a	agua
8	**l**	Lateral, ápico-gingival, sonora	**l**oBo	lobo
9	**m**	Nasal, bilabial, sonora	**m**esa	mesa
10	**m**	Nasal, labiodental, sonora	e**m**fermo	enfermo
	n	Nasal, ápico-gingival, sonora	**n**ada	nada
11	**N**	Nasal, velar, sonora	o**N**Go	hongo
12	**J**	Nasal, dorso-palatal, sonora	ni**J**o	niño
13	**B**	Fricativa, bilabial, sonora	tu**B**o	tubo
14	**D**	Fricativa, apico-interdental, sonora	a**D**a	hada
15	**G**	Fricativa, dorso-velar, sonora	a**G**wa	agua
16	**b**	Oclusiva, bilabial, sonora	**b**eso	beso
17	**d**	Oclusiva, ápico-dental, sonora	**d**ar	dar
18	**g**	Oclusiva, dorso-velar, sonora	**g**ula	gula
19	**r**	Vibrante simple, ápico-gingival, sonora	pe**r**o	pero
20	**R**	Vibrante múltiple, ápico-gingival, sonora	ka**R**o	carro
	R	Vibrante múltiple-fricativa, apical, sonora	ka**R**o	carro
21	**L**	Lateral, dorso-prepalatal, sonora	**L**uBja	lluvia
	Z	Fricativa, dorso-prepalatal, sonora	**Z**uBja	lluvia
	Z	Africada, dorso-prepalatal, sonora	Kon**Z**uGe	cónyuge
	Z	Fricativa, dorso-prepalatal, sorda	**Z**ubja	lluvia
	Z	Fricativa, dorso-palatal, sonora	**Z**ubja	lluvia
22	**h**	Fricativa, laríngea, sorda	a**h**ta	hasta
23	**p**	Oclusiva, bilabial, sorda	**p**ala	pala
24	**t**	Oclusiva, ápico-dental, sorda	**t**ieRa	tierra
25	**k**	Oclusiva, dorso-velar, sorda	**k**ilo	kilo
26	**H**	Africada, dorso-prepalatal, sorda	te**H**o	techo
27	**s**	Fricativa, dorso-gingival, sorda	**s**ala	sala
28	**f**	Fricativa, labio-dental, sorda	**f**e	fe
29	**x**	Fricativa, dorso-velar, sorda	**x**wes	juez
30	**C**	Fricativa, dorso-palatal, sorda	ar**C**entina	argentina

Tabla 3.1: *Alfabeto SAMPA para el español de la Argentina, extraído de [9]*

3.1.1. Inventario fonético utilizado en la creación de voces

En el alfabeto fonético presentado anteriormente se distinguen los alófonos de algunos fonemas. Como ya explicamos, los ALÓFONOS son conjuntos de fonos que en un idioma se reconocen como el mismo fonema. Por ejemplo, el fonema /n/ tiene alófonos [n] y [N], o sea dos instanciaciones distintas para la misma clase abstracta /n/ del español. Estos alófonos están presentes, por ejemplo, en las palabras [nada] (nada) y [oNGo] (hongo).

En este trabajo se decidió descartar los alófonos y utilizar sólo fonemas (clases abstractas de sonidos) y las versiones acentuadas de las vocales. La elección de un alófono u otro debería darse por el contexto de la palabra en que ocurren. Es, entonces, tarea de los algoritmos de selección de unidades priorizar y optar por fonemas con contexto similar al fonema objetivo. Esto se logra gracias a que cada unidad en la base de datos cuenta con un conjunto asociado de características de contexto e intrínsecos a cada unidad, lo que permite la comparación. Luego, los símbolos omitidos y reemplazados por su correspondiente clase abstracta fueron: N, B, D, G, h y C.[1]

La tabla 3.2 muestra los fonemas utilizados en la construcción de todas las voces a lo largo del trabajo junto con sus características articulatorias. Para más información sobre las características de los fonemas presentes en la tabla, ver anexo A.

[1] Debido a una omisión involuntaria se conservaron los fonos Z (alófono de L), j (alófono de i) y w (alófono de u), pese a que debieron ser retirados también.

		vlng	vheight	vfront	vrnd	ctype	cplace	cvox
Vocales	a	l	3	1	−	0	0	−
	e	l	2	1	−	0	0	−
	i	l	1	1	−	0	0	−
	o	l	3	3	+	0	0	−
	u	l	1	3	+	0	0	−
	j	d	1	1	−	0	0	−
	w	d	1	3	+	0	0	−
Vocales acentuadas	a1	l	3	2	−	0	0	−
	e1	l	2	1	−	0	0	−
	i1	l	1	1	−	0	0	−
	o1	l	2	3	+	0	0	−
	u1	l	1	3	+	0	0	−
Consonantes	m	0	−	−	−	n	l	+
	n	0	−	−	−	n	a	+
	J	0	−	−	−	n	p	+
	s	0	−	−	−	f	a	−
	f	0	−	−	−	f	b	−
	x	0	−	−	−	f	v	−
	Z	0	−	−	−	f	p	+
	b	0	−	−	−	s	l	+
	d	0	−	−	−	s	d	+
	g	0	−	−	−	s	v	+
	p	0	−	−	−	s	l	−
	t	0	−	−	−	s	d	−
	k	0	−	−	−	s	v	−
	l	0	−	−	−	l	a	+
	r	0	−	−	−	l	a	+
	R	0	−	−	−	l	a	+
	L	0	−	−	−	l	p	+
	H	0	−	−	−	a	p	−
Silencio	#	0	−	−	−	0	0	−

Tabla 3.2: *Inventario fonético utilizado para la creación de voces; donde vleng (vowel length) denota duración de la vocal pudiendo ser {s:short, l:long, d:dipthong, a:schwa, 0}; vheight (vowel height) denota la altura de la vocal y puede tomar los valores {1, 2, 3, −}; vfrton (vowel frontness) denota la localización vocálica y toma los valores {1, 2, 3 −}; vrnd (lip rounding) denota el redondeamiento labial {+, −}; ctype (consonant type) denota el tipo de consonante y puede ser {s:stop, f:fricattve, a:affricative, n:nasal, l:liquid, 0:no aplica}; cplace (consonant place) denota la ubicación de articulación, siendo posible {l:labial, a:alveolar, p:palatal, b:labio-dental, d:dental, v:velar, 0}; y por último cvox (consonant voicing) denota la sonoridad de la consonante, es decir, si las cuerdas vocales vibran o no al articular la consonante {+, −}.*

Capítulo 4

Desarrollo del trabajo

A continuación se describen los problemas encontrados durante el desarrollo del trabajo, así como también las soluciones empleadas.

4.1. Modificaciones realizadas al cuerpo de datos SECYT

4.1.1. Etiquetados

Durante la creación de las voces, fueron detectados varios errores de etiquetado en la base SECYT, tanto en archivos .fon como en .gra. Por este motivo, en la primera voz creada se pudieron utilizar solamente 578 archivos de los 741 para armar la base de datos de unidades. Se procedió entonces a la corrección de los 163 archivos erróneos para incorporarlos finalmente al inventario. Para ello, se cruzó información de los dos tipos de archivos .gra y .fon. En ellos, cada línea representa un fonema con alineación temporal en .fon, o bien una palabra con alineación temporal en .gra.

Dentro de los problemas existentes se encontraron líneas desordenadas temporalmente, tiempos y fonemas etiquetados de manera errónea, inconsistencias entre archivos .fon y .gra, líneas de más, líneas faltantes y transcripciones no concordantes con los audios (fonos presentes en los audios pero faltantes en las transcripciones, por ejemplo la palabra "inscripción" fue etiquetada como /ihkripsjon/, cuando en realidad la /n/ si era pronunciada por la locutora y debería ser etiquetada como /inhkripsjion/).

En los casos en que faltaba algún fonema en el etiquetado, se utilizó el programa *Praat*[1], una herramienta gratuita para el análisis acústico del habla, para obtener la información faltante y completar el etiquetado manualmente.

Una vez corregidos todos los archivos se volvió a crear la voz, y los resultados fueron notoriamente mejores al utilizar mayor cantidad de unidades dentro de la base.

4.1.2. Cobertura de difonos

Independientemente del tipo de unidad elegido para la síntesis, es importante que los textos utilizados para la posterior generación de la base de datos cubran la mayor cantidad de posibles combinaciones fonéticas. Esto se debe a que la coarticulación entre los fonos (es decir, la influencia de un fono sobre sus vecinos) es un factor que jugará un rol importante en la inteligibilidad y naturalidad del habla sintetizada.

Al crear una voz con Festival TTS, utilizando unidades similares a difonos, quedó en evidencia la cantidad de combinaciones faltantes en el corpus SECYT. En el inventario fonético hay 31 fonos, contando el silencio. En teoría, para tener un mapeo completo, debería haber $31^2 = 961$ combinaciones; sin embargo, en la base de datos sólo existen 645, dando como resultado un faltante de 316 difonos. No obstante, hay que tener en cuenta que muchas de las combinaciones faltantes no tienen sentido en el español, como por ejemplo la combinación "qk". Ver anexo B para listado completo de difonos faltantes.

Lamentablemente, la única solución a este problema es la creación y posterior grabación de un conjunto de oraciones que cubra todas las combinaciones de posibles difonos presentes en el idioma. No es una tarea sencilla ni rápida, y queda fuera del alcance de este trabajo. Por este motivo, ciertas palabras no podrán ser sintetizadas por las voces creadas por concatenación de difonos.

4.1.3. Acentos léxicos

Los acentos léxicos son una característica de la lengua que nos permite, entre otras cosas, diferenciar y desambiguar palabras. Si bien en las trans-

[1] http://tcts.fpms.ac.be/synthesis/

cripciones de las oraciones del corpus SECYT los acentos léxicos estaban presentes, no había un símbolo o caracter especial que denotara una vocal acentuada en las transcripciones fonéticas de las mismas. Por ende, al no haber una forma de diferenciar una vocal acentuada de una no acentuada, en la base de datos de sonidos aparecen bajo la misma etiqueta (por ejemplo, no se distinguirían la primera y la segunda /a/ de /kasa/). Eso no debería ocurrir, ya que al concatenar las unidades se obtendrá un resultado no deseado.

Para resolver este problema, se combinaron las transcripciones fonéticas generadas automáticamente como resultado de uno de los pasos del proceso de creación de voces de Festival, con las provistas por el LIS. Las transcripciones fonéticas automáticas se generaron a partir de las transcripciones manuales de los audios y la aplicación de reglas letra-a-sonido que fueron escritas específicamente para este trabajo, siguiendo las reglas fonéticas del español. Esto fue posible gracias a que, a diferencia de otros lenguajes, la ortografía del español tiene un mapeo casi directo con la pronunciación de las palabras.

4.1.4. Grabaciones

Otro factor que degradaba la calidad de las voces construidas era el modo en el que los audios fueron grabados. Cuando uno construye un sistema TTS y no posee muchas instancias de cada sonido, es indispensable que los audios estén grabados de una forma monótona o pareja, sin grandes cambios de intensidad, tono o duración. Así, al concatenar las unidades, se evitará generar ruidos (clics) en las uniones causados por discontinuidades de fase, tono o espectro.

Usualmente es un hecho deseable que los audios utilizados en sistemas texto-a-habla sean grabados por locutores, ya que ellos están entrenados para modular la voz y pronunciar correctamente. Sin embargo, también es cierto que se debe tener especial cuidado en no generar grandes discontinuidades en la prosodia y cambios en la entonación, características que también son intrínsecas al rol del locutor.

Como las oraciones de SECYT fueron grabadas por una locutora argentina para realizar un posterior estudio de la prosodia del habla y no para construir un sistema TTS, muchos de los aspectos mencionados anteriormente no fueron tenidos en cuenta durante la realización de la tarea, lo que provocó que el resultado no fuera óptimo para ser empleado en un sistema

TTS. Esto se pudo apreciar en las primeras voces creadas, en donde, lejos de sonar naturales, se escuchaban ruidos y cambios tonales constantes, lo que provocaba un efecto desagradable al oído.

Para atenuar esos efectos no deseables en la síntesis, se emplearon técnicas de procesamiento de señales sobre los audios originales, buscando nivelar la intensidad entre ellos y reducir los picos tonales generados por la locutora. Para esta tarea se utilizó la herramienta Praat.

Nivelación de la intensidad

Con respecto a la intensidad media de los audios, los mismos presentaban una variabilidad entre un mínimo de 47.89 db y un máximo de 64.36 db, siendo 58.41 db la media y 2.36 db el desvío standard. Para contrarrestar el efecto sonoro que esta variabilidad producía en los audios sintetizados, la intensidad se niveló a 72 db por audio usando interpolación lineal.

Reducción de picos tonales

Para reducir los picos tonales presentes en los audios, se trabajó sobre el contorno de F0 (*pitch track*) de cada uno, en el cual se puede observar la estimación de la frecuencia fundamental a lo largo del tiempo. Praat brinda la posibilidad de extraer el contorno de F0 de un audio y modificar cada punto del mismo según una función de entrada.

Para la manipulación del contorno de F0, Praat utiliza una técnica conocida como TD-PSOLA (*Time-Domain Pitch-Synchronous OverLap-and-Add*) [12]. La misma consiste en dividir la señal en ventanas alrededor de cada *pitch mark* y recombinarlos para modificar la prosodia del audio, ya sea en duración o nivel tonal. La idea es, básicamente, identificar ciclos en la señal. La nueva forma de onda es creada simplemente solapando o agregando ciclos. Para cambiar la duración se agregan o borran ciclos, y para cambiar el tono, se juntan o separan ciclos.

Se realizaron pruebas con varias fórmulas sobre los audios originales y se optó por la que se considera que dio mejores resultados[2]. Como primera aproximación para resolver el problema, se utilizó la siguiente fórmula:

$$\text{if } (p_i > u) \text{ then } ((p_i - u) * k + u) \text{ else } (p_i) \text{ fi} \tag{4.1}$$

[2]Evaluados subjetivamente por la autora del presente trabajo.

donde u es un umbral (en Hz) y k es una constante de reducción del nivel tonal.

La fórmula (4.1) simplemente escala, mediante la multiplicación de un coeficiente k, todo punto que supere el umbral u. Se realizaron pruebas variando los valores de k (0.1, 0.2, 0.3, 0.4) y u (150 ó 200).

También se realizaron variaciones a la fórmula inicial, sumando o multiplicando una constante al resultado de la misma para subir de tono toda la señal, ya que al modificarla, la voz de la locutora quedaba muy apagada.

Las funciones descritas anteriormente generaban variaciones abruptas cerca del umbral u, lo que no impedía que se continuara percibiendo el cambio de nivel tonal. Para atenuar este problema y lograr que las variaciones fueran más suaves, se recurrió a una función polinómica de grado 2. La misma fue construida utilizando aproximación por cuadrados mínimos a partir de valores fijados de antemano en una tabla (ver tabla 4.1).

Valores de entrada del nivel tonal	Valores deseados de salida del nivel tonal
0	0
50	50
100	100
125	120
170	145
200	160
300	180

Tabla 4.1: *Tabla de conversión de tono*

Luego de varios ajustes en los valores de la tabla, la fórmula que mejores resultados audibles dio fue la siguiente.

$$-0{,}002 * p_i^2 + 1{,}2092 * p_i + 48{,}315 \tag{4.2}$$

El término independiente se ajustó para subir de tono toda la señal, además de reducir los picos tonales.

La figura 4.1 muestra las curvas que grafican las funciones probadas para reducir los picos tonales presentes en los audios. Notar que la curva fucsia, resultado de la función (4.2), es más suave en relación al resto. La figura 4.2,

exhibe dos ejemplos de reducción de picos tonales. En cada uno se muestra el contorno original de F0 de un audio, junto con el contorno de F0 resultante de haber aplicado la transformación (4.2).

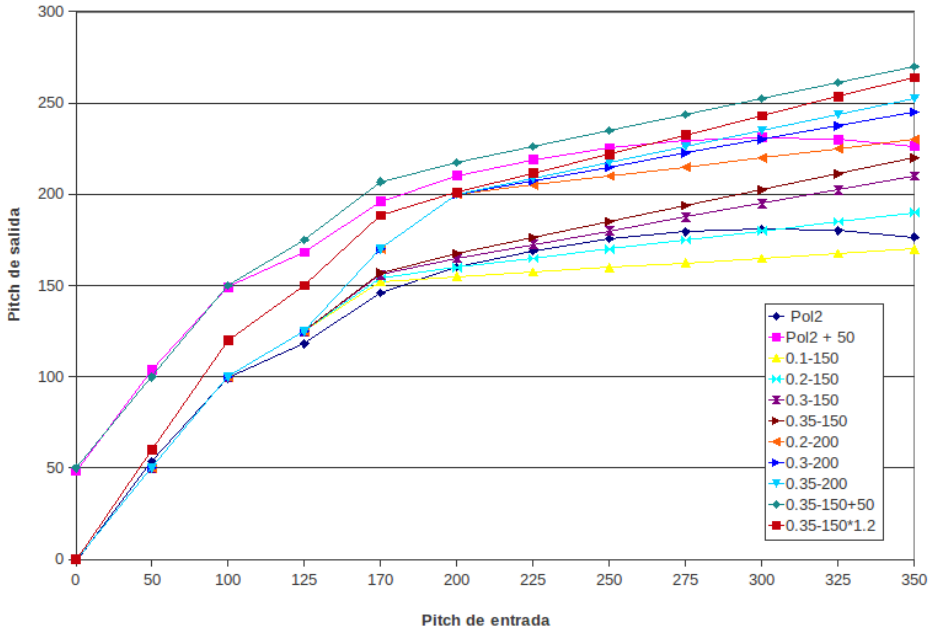

Figura 4.1: *Funciones para reducir picos tonales*

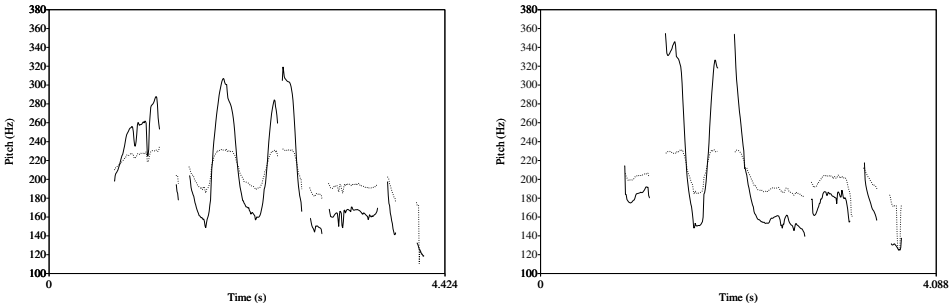

Figura 4.2: *Ejemplo - Reducción de picos tonales*

Se ajustaron los valores mínimo y máximo de F0 en 130 Hz y 380 Hz respectivamente, según resultados obtenidos en la tesis de licenciatura de Lautaro Dolberg [7] sobre el rango de F0 de la locutora que grabó los audios del corpus SECYT. Estos valores se usaron en todas las estimaciones de parámetros acústicos realizadas en esta tesis.

4.1.5. Otras modificaciones menores

En muchos casos, dentro de las transcripciones fonéticas del LIS, se encontraron fonemas "fusionados". Es decir, fonemas idénticos que deberían aparecer contiguos, aparecían una sola vez. Por ejemplo, las palabras "cientos de edificios" aparecían transcriptas en SECYT como "s j e n t o s d e d i f i s j o s", con la segunda y tercera 'e' fusionadas en una sola. Este problema fue solucionado en forma manual.

Como se mencionó en la Subsección 4.1.3, las transcripciones fonéticas creadas automáticamente en Festival se crearon a partir de los audios, sus transcripciones y reglas letra-a-sonido. En las transcripciones ortográficas de los audios no había un símbolo que indicara las pausas largas entre palabras. Luego, tampoco quedaron marcadas en las transcripciones fonéticas automáticas. En cambio, en las transcripciones fonéticas las pausas estaban denotadas con '/p'. El problema que origina no tener etiquetadas las pausas es que, al recortar las unidades para crear la base de datos de unidades, los silencios quedan pegados a los fonemas vecinos, provocando que posteriormente aparezcan en el habla sintetizada. La solución fue agregar una ',' en las transcripciones de los audios, en los lugares donde apareciera el símbolo '/p' en las transcripciones fonéticas manuales. La ',' es uno de los símbolos que utiliza Festival para introducir pausas.

Además de los cambios mencionados anteriormente, también se eliminaron de las transcripciones las letras mudas, como por ejemplo la "p" de la palabra "ptolemaica".

4.2. Creación de voces con Festival TTS

Festival Speech Synthesis System fue el primer sistema considerado para comenzar a desarrollar las voces, ya que entre todas las herramientas de código abierto para creación de voces, es la más difundida.

Inicialmente se creó un sistema básico de síntesis de dominio limitado

para decir la hora en inglés, con el fin de familiarizarse con las herramientas disponibles y lograr un mejor entendimiento del proceso de creación de voces. En los sintetizadores de dominio limitado las unidades suelen ser palabras o frases. Este vocabulario específico o frases con las que se crea el sintetizador, restringen el habla sintetizada. Las oraciones utilizadas para armar la base de datos de unidades seguían el siguiente formato:

The time is now, EXACTNESS MINUTE INFO, in the DAYPART

Un ejemplo podría ser: "The time is now, exactly five past one, in the morning".

El siguiente paso fue la creación del nuevo lenguaje "spanish_arg". Se utilizó como guía el lenguaje "spanish" utilizado por una voz ya existente en Festival llamada "el_diphone", una voz de difonos para español ibérico.

Para crear un lenguaje y posteriormente una voz, es fundamental e indispensable la definición de un conjunto de fonos (phoneset), y un módulo que provea la pronunciación de palabras (lexicon). Este módulo contiene un diccionario fonético de palabras y un método para pronunciar palabras desconocidas, palabras que no se encuentren en el diccionario.

El español es un lenguaje cuya pronunciación puede ser predicha, casi por completo, mediante su ortografía. Luego, no es necesario tener un diccionario de palabra-pronunciación, sino que se puede hacer la mayor parte del trabajo con reglas letra-a-sonido.

Se crearon, entonces, los archivos lexicon.scm y phoneset.scm correspondientes al lenguaje spanish_arg, conteniendo las reglas LTS (reglas de letra-a-sonido) y el conjunto de fonemas respectivamente. Para la construcción de las reglas LTS, se tuvieron en cuenta las reglas fonológicas del español. En la Sección 3.1 se puede encontrar información detallada del phoneset utilizado.

Todas las voces descriptas en las próximas secciones se construyeron en base al lenguaje *spanish_ arg* creado.

4.2.1. Clunits

Cluster unit selection (Clunits)[3] es una técnica de concatenación de unidades. La idea es agrupar cada tipo de unidades presentes en la base de

[3] http://festvox.org/festvox/x3082.html

datos, en conjuntos (clusters) de unidades acústicamente similares, basándose en información no acústica disponible al momento de la síntesis, como ser contexto fonético, características prosódicas (F0 y duración), y otras características de más alto nivel como el énfasis, la posición de las palabras y los acentos.

La predicción de la prosodia (contorno entonacional, F0, etc.) de las unidades puede ser dividida en dos tareas:

1. Predicción de acentos (y/o tonos): Esto se hace sílaba por sílaba, identificando qué sílabas se acentúan, así como también qué tipo de acento se requiere (primario, secundario, etc.),
2. Generación de un contorno de F0: Generar un contorno de F0 a partir de los acentos y los tonos.

Para el paso (1.) se reusó un árbol de decisión, encontrado en la documentación de Festival, para acentuar sílabas tónicas en palabras de contenido (sustantivos, verbos, adjetivos y la mayoría de los adverbios); y para el paso (2.) se utilizó una función empleada en la voz el_diphone.

Para asignar duración a cada segmento, se utilizó un CART (*Classification and Regression Tree*) para el español, utilizado como ejemplo en la documentación de Festival. Los CART se entrenan a partir de una base de datos de habla natural, derivando los factores mediante métodos estadísticos. Un árbol simple de decisión de duración, predice el factor para modificar la duración promedio de un segmento. Este árbol causa la prolongación del inicio y final de oraciones teniendo en cuenta el énfasis. En el anexo C, se puede encontrar el código Scheme de todas las funciones mencionadas en esta sección.

Unifonos

Las primeras voces construidas son del tipo al que llamaremos UNIF (unifonos[4]). Las mismas se obtuvieron como resultado del proceso de creación de voces Clunits, detallado en la documentación de Festival, utilizando *fonos* como tipo de unidad y los audios SECYT originales.

Luego de varios intentos y pruebas, modificando parámetros y funciones dentro de los módulos aplicados en el proceso de síntesis, se eligió la configu-

[4]La palabra "unifono" es simplemente otra forma de referirse a un fono.

ración que se considera dio mejores resultados audibles[5]. Sin embargo, esos resultados no llegaron a cumplir las expectativas en cuanto a calidad e inteligibilidad.

El notorio cambio de tono y volumen en los audios sintetizados, dejó en evidencia las características problemáticas de las grabaciones descriptas en la Subsección 4.1.4. Como se menciona en dicha sección, la solución empleada fue nivelar la intensidad entre los audios y reducir los picos tonales mediante la aplicación de una función polinomial a cada *pitch mark*. Haremos referencia a estos nuevos audios, como *audios modificados*.

Una vez modificados los audios de SECYT, se volvió a construir la voz Clunits. Las mejoras en los audios sintetizados con esta nueva versión, parecían notorios. Se hará referencia a estos sistemas con los nombres de UNIF_ORIG y UNIF_MOD, siendo ORIG y MOD el estado de los audios utilizados.

Difonos

Recordemos que un difono es el sonido que queda delimitado por la región estable de un fonema hasta la región estable del siguiente. La construcción de una voz utilizando esta definición de difonos no fue posible, debido a que no encontramos la manera de compatibilizar el formato existente del etiquetado de los datos con el etiquetado requerido. Resolverlo no era sólo una cuestión de compatibilizar notación, sino que había que modificar toda la estructura y volver a etiquetar manualmente los datos, alineando temporalmente difonos en lugar de fonos. Las herramientas y scripts utilizados en Festival requieren que las oraciones portadoras utilizadas para grabar y etiquetar los difonos cuenten, además, con una identificación de archivo y la lista de difonos presentes en dicha oración, según el siguiente formato:

(uk_0001 “pau t aa b aa b aa pau” (“b-aa” “aa-b”))

Como aproximación, se intentó “simular” un difono utilizando *fono_anterior + fono*. Se esperaba que esta nueva unidad, a pesar de no ser un difono puro, capturara la coarticulación (influencia de un fono sobre sus vecinos) y mejorara los saltos en la señal sintetizada. Cabe aclarar, que en este caso la unidad que se recorta y concatena sigue siendo un solo fono. A la hora de seleccionar los fonos que se utilizarán para la síntesis, se toma en cuenta el

[5]La evaluación fue realizada en forma subjetiva por la autora

fono anterior al calcular las ventajas y desventajas de cada candidato. Es decir, si dos fonos candidatos son idénticos, desempata el antecesor de cada uno en la base de datos: gana el antecesor más parecido al *fono anterior* en cuestión.

Con este nuevo tipo de unidades DIF, se crearon dos voces Clunits, una en base a los audios originales y otra en base a los audios modificados. A estos sistemas los llamaremos DIF_ORIG y DIF_MOD. Consideramos que en ambos casos la calidad aumentó. Sin embargo, en estas voces está presente el problema de los difonos faltantes, mencionado en la sección anterior, que provoca que algunas palabras no puedan sintetizarse.

4.2.2. CLUSTERGEN - Síntesis estadística paramétrica

CLUSTERGEN [2] es un método para construir sintetizadores estadísticos paramétricos a partir de bases de datos de habla natural. Se entrenan modelos paramétricos a partir de los datos disponibles en la base, y se utilizan al momento de la síntesis.[6]

La síntesis estadística paramétrica, también conocida como síntesis basada en generación de HMM, tiene la ventaja de suavizar los datos. El posible número de combinaciones de segmentos en la síntesis concatenativa suele ser vasto, y algunas concatenaciones pueden introducir malas uniones. Testear esto es bastante complicado, y solucionar los problemas no es una tarea simple. La síntesis estadística paramétrica aborda este problema mediante la construcción de lo que puede ser visto como un promedio de unidades, en lugar de utilizarse un conjunto de instancias, como en el caso de la síntesis concatenativa.

Sin embargo, hay algunas desventajas con este método. La técnica requiere una parametrización del habla que sea reversible y tenga propiedades modelables, como por ejemplo, con una distribución gaussiana. Una de esas parametrizaciones, usada en Festival, es MFCC (*Mel Frequency Cepstral Coefficients*)[6]. Al igual que con muchas parametrizaciones del habla sin un modelo explícito de excitación, el habla sintetizada resulta robótica y tiene un zumbido poco natural, que carece de la nitidez y frescura que se suele encontrar en sintetizadores por selección de unidades.

Utilizando este método, se crearon dos voces HMM. Nuevamente, una en base a los audios originales y otra en base a los audios modificados. Se hará

[6] http://festvox.org/bsv/c3170.html

referencia a estos sistemas con los nombres de HMM_ORIG y HMM_MOD.

4.3. Creación de voces con MARY TTS

El sistema MARY TTS fue construido en base a Festival, fue posible reciclar gran parte de la información requerida para la creación del lenguaje en español argentino y las voces, sin mayores dificultades. Para la creación de las voces en MARY, se utilizaron directamente los audios de SECYT modificados. Se crearon dos voces con MARY: una de concatenación de difonos y una basada en HMM.

4.3.1. Difonos

El sistema de selección de unidades en MARY implementa un algoritmo de selección de unidades genérico, combinando los pasos habituales de preselección de unidades candidatas basándose en árboles de decisión, una fase de programación dinámica que combina costos ponderados de unión y costos objetivos, y una fase de concantenación en donde se unen las unidades seleccionadas generando el audio de salida. Se utilizan difonos como unidades en lugar de fonos, ya que se espera que la unión en la sección media de fonemas introduzca menos discontinuidades y ruido que realizar las uniones en los límites de los fonemas.

Para cada difono objetivo, se selecciona un conjunto de unidades candidatas, eligiendo por separado, candidatos para cada semifono (*half-phone*) a través de árboles de decisión y quedándose sólo con aquellos que son parte del difono requerido. Cuando no hay difonos que coincidan con la búsqueda, el sistema hace un fallback a semifonos. Para una descripción mas detallada sobre el proceso de selección de unidades en MARY, ver [16].

Para poder crear la primera voz, al igual que en Festival, fue necesaria la creación de un módulo para el lenguaje español. Sin embargo, a diferencia de Festival, en MARY no se pueden especificar a mano las reglas LTS, sino que se generan mediante aprendizaje automático utilizando un diccionario de "palabra-transcripción fonética" requerido antes del comienzo del proceso de creación de un lenguaje. Luego, creamos dicho diccionario utilizando las palabras y transcripciones fonéticas de todas las oraciones SECYT.

A pesar de obtener buenos resultados respecto a la calidad del sonido en la nueva voz creada, a la que llamaremos MARY_DIF, la asignación letra-a-

sonido no era la correcta. Por ejemplo, la palabra “que” era transcripta con la secuencia de fonemas “k u e” en lugar de “k e”.

Sin embargo, como el problema era justamente la asignación letra-a-sonido con las reglas generadas automáticamente, y no un problema de recorte de unidades o concatenación, utilizamos una de las particularidades de MARY que permite al usuario observar el resultado de alguno de los pasos intermedios del proceso de síntesis, modificarlo y luego reanudar el proceso partir de ese punto.

La modularidad del sistema y la representación interna de los datos, posibilitan al usuario comenzar el proceso de síntesis ingresando diferentes tipos de datos, diferentes representaciones de lo que se quiere sintetizar. Tres formas distintas de sintetizar el mismo texto podrían obtenerse ingresando:

- texto plano (modo TEXT)

  Bienvenido al mundo de la síntesis del habla,

- secuencia de fonemas (modo SIMPLEPHONEMES):

  bien-be-’ni-do al ’mun-do de la ’sin-te-sis del ’a-bla

- secuencias de fonemas con información adicional según el siguiente formato (modo SIMPLEPHONEMES):

```
?xml version="1.0" encoding="UTF-8"?>
<maryxml xmlns="http://mary.dfki.de/2002/MaryXML"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 version="0.5" xml:lang="es">
<p>
<s>
<t g2p_method="rules" ph="b i e n - b e - ’ n i - d o"
 pos="content">
Bienvenido
</t>
<t g2p_method="userdict" ph="a l" pos="function">
al
</t>
<t g2p_method="userdict" ph="’m u n - d o" pos="function">
mundo
</t>
<t g2p_method="userdict" ph="d e" pos="function">
```

```
de
</t>
<t g2p_method="userdict" ph="l a" pos="function">
la
</t>
<t g2p_method="rules" ph="’ s i n - t e - s i s" pos="function">
síntesis
</t>
<t g2p_method="userdict" ph="d e l" pos="function">
del
</t>
<t g2p_method="userdict" ph="’ a - b l a" pos="function">
habla
</t>
</s>
</p>
</maryxml>
```

Luego, una solución más directa que la empleada (ingresar texto plano, ver el resultado en tipo PHONEMES, modificarlo según lo deseado y generar el audio a partir de eso), habría sido ingresar los datos en tipo SIMPLE-PHONEMES, es decir, ingresar directamente los fonemas. Sin embargo, esta opción no fue factible debido a una mala programación de esa funcionalidad. Se intentó resolver el problema, pero la mala documentación y la gran cantidad de código y módulos existentes, impidieron la tarea.

Se instalaron varias versiones de MARY, incluyendo un release de Junio de 2012, y todas presentan fallas en esa funcionalidad, entre otras. No son versiones estables y tampoco hay disponible buena documentación que sirva de guía y soporte para intentar solucionar los problemas que se detectan. En consecuencia, los resultados obtenidos fueron generados manualmente. Será necesario trabajar a futuro para resolver los problemas técnicos de MARY de modo de poder automatizar el proceso de síntesis.

4.3.2. HMM

Al igual que Festival, para crear voces basadas en HMM, la plataforma MARY utiliza los scripts provistos por HTS (*HMM-based Speech Synthesis System*)[7]. Los scripts y programas usados para entrenar las voces HMM en MARY fueron ligeramente modificados. Básicamente se modificaron para

[7] http://hts.sp.nitech.ac.jp/

que utilicen los features de contexto predichos por el analizador de texto de MARY en lugar del de Festival. Utilizando este método, se construyó la voz a la que denominaremos MARY_HMM.

Esta versión presenta otro problema además de la aplicación de reglas equívocas de LTS. Como se puede observar en el ejemplo PHONEMES de la sección anterior, uno de los módulos es el encargado de silabificar las palabras (sílabas separadas por "-"). No se encontró el motivo, pero al sintetizar un texto con la voz MARY_HMM, el sistema realiza una pausa después de cada sílaba. Por esta razón, para obtener la síntesis deseada, fue necesario corregir los fonemas mal transcriptos y borrar los "-" al pasar por el paso intermedio PHONEMES.

Nuevamente, los resultados obtenidos fueron generados en forma manual. Queda como trabajo futuro estudiar cómo sortear los problemas de MARY TTS y automatizar ese proceso.

Capítulo 5

Evaluación de los sistemas

Resulta difícil evaluar sistemas texto a habla ya que definir cuán bien suena uno, es una tarea subjetiva que varía fuertemente en función del oyente. De todas formas, existen evaluaciones subjetivas que nos acercan a cumplir el objetivo. Para ello se pueden tener en cuenta dos dimensiones: *naturalidad* e *inteligibilidad*.

Un test de naturalidad mide la calidad de un sistema TTS: cuán natural suena. En cambio, un test de inteligibilidad mide la capacidad de los oyentes para identificar las palabras escuchadas. Obviamente la inteligibilidad afecta la naturalidad de un sistema. Sin embargo, un sistema puede ser perfectamente inteligible y a la vez ser poco natural y desagradable al oído.

5.1. Tests y diseño experimental

Existen evaluaciones estándares para medir la calidad general de sistemas TTS, así como también tests específicos para evaluar prosodia, inteligibilidad y naturalidad, entre otros. Los experimentos llevados a cabo en este trabajo, se basaron en el trabajo realizado por Gurlekian et al. (2012). En dicho trabajo, se evalúan y comparan tres TTS por concatenación de unidades para el idioma español argentino, utilizando diferentes tests de percepción.

Para evaluar la naturalidad, se utilizó la escala *MOS* (*The Standard Mean Opinion Score*), un método en el cual los oyentes usan una escala fija del 1 al 10 para evaluar la calidad global de un sistema [14] [20]. Es un test de propósito general que provee puntajes promedio para habla natural y

artificial.

Se agregaron 20 oraciones a las utilizadas en [10], debido a la cantidad de sistemas evaluados en este caso (en los experimentos del LIS sólo participaron tres sistemas).

Para medir la inteligibilidad se utilizó el método *SUS* (*Syntactically Unexpected Sentences*), conocido por ser el más estricto para evaluar inteligibilidad [13]. Las oraciones utilizadas en SUS, se caracterizan por ser sintácticamente correctas pero sin significado o muy baja probabilidad de ocurrencia.

Se evaluaron las voces construidos con Festival (UNIF_ORIG, UNIF_MOD, DIF_ORIG, DIF_MOD, HMM_ORIG, HMM_MOD) y MARY (MARY_DIF, MARY_HMM) en forma simultánea (ver tabla 5.1). Un total de 20 oyentes sin problemas auditivos, de entre 21 y 61 años, participaron en el experimento MOS, y 10 en el experimento SUS. A cada participante se

	Festival TTS			**MARY TTS**	
	Unifonos	Difonos	HMM	Difonos	HMM
Audios originales	UNIF_ORIG	DIF_ORIG	HMM_ORIG	-	-
Audios modificados	UNIF_MOD	DIF_MOD	HMM_MOD	MARY_DIF	MARY_HMM

Tabla 5.1: *Sistemas construidos*

le entregaron las instrucciones en papel, junto con el lugar destinado a las respuestas.

5.2. Test MOS (*Mean Opinion Score*)

Para el test MOS se sintetizaron 320 textos (40 oraciones MOS * 8 voces) de entre 5 y 20 palabras, con dos o tres frases melódicas largas. Para cada oyente, 40 oraciones fueron presentadas (5 de cada voz), elegidas y ordenadas al azar, pudiendo haber repeticiones entre voces pero no dentro de la misma voz. Las instrucciones dadas a cada participante fueron:

> Puntúe la calidad (naturalidad) de lo escuchado en base a una escala del 1 al 10, donde 1 significa: "No suena natural en lo absoluto", y 10 significa: "Suena completamente natural". Podrá escuchar cada audio la cantidad de veces que lo considere necesario.

Un ejemplo de las oraciones utilizadas en este test es: "El sector de informática es el nuevo generador de empleo del país". Ver anexo D para el listado completo de oraciones.

5.3. Test SUS (*Syntactically Unexpected Sentences*)

Se emplearon 50 textos diseñados sin sentido semántico pero con correcta estructura sintáctica, cada uno con una longitud de entre 6 a 10 palabras y conteniendo una o dos frases melódicas. Para cada oyente se eligieron 40 oraciones al azar (5 de cada voz), sin repeticiones. Las instrucciones dadas a cada participante fueron:

> Escriba cada palabra que escuche en cada oración. Preste atención ya que no habrá repeticiones.

Un ejemplo de oración utilizada en SUS, sin sentido semántico pero correcta estructuralmente es: "El viento dulce armó un libro de panqueques". Ver anexo D para el listado completo de oraciones.

5.4. Resultados

Como se mencionó en la sección anterior, para medir naturalidad e inteligibilidad se utilizaron los tests MOS y SUS. El principal objetivo de estas evaluaciones, además de intentar responder qué voz resultó ser la mejor en términos generales, es analizar si las técnicas utilizadas en este trabajo para mejorar los audios a partir de los cuales se construyeron los sistemas TTS, implicaron una mejora en el habla sintetizada, o no.

Recordar que se estarán evaluando ocho voces en simultáneo: seis construidas con Festival y dos con MARY TTS. Los audios originales (ORIG) son las grabaciones originales del corpus SECYT. Cuando se hace referencia a audios modificados (MOD), se habla de los audios resultantes de haber aplicado la función de reducción de picos tonales y de nivelación de intensidad, a los audios originales (ver tabla 5.1 y Subsección 4.1.4).

A lo largo de esta sección, entonces, se tratará de responder a las siguientes preguntas, entre otras:

P1. ¿Cuál es la mejor voz?, donde "mejor" dependerá de lo que se esté evaluando en cada caso.

P2. ¿Las voces MOD son mejores que las voces ORIG? ¿En qué casos?

P3. ¿Cuál de las tres voces construidas en base a Festival es mejor (UNIF, DIF o HMM)?

P4. ¿Cuál de las dos voces de MARY TTS es mejor (DIF o HMM)?

P5. ¿Cuál de las dos plataformas, MARY TTS o Festival, resultó mejor para la creación de cada tipo de voz?

5.4.1. Test de naturalidad MOS

Para reducir los efectos de interpretación de la escala MOS, todos los datos se normalizaron por hablante mediante z-scores[1]. La figura 5.1, muestra la distribución de los puntajes MOS normalizados obtenida para cada voz, mientras que en la figura 5.2 se observa lo mismo pero con los datos sin normalizar. Se puede apreciar que las voces que presentan mayor calidad general son MARY_HMM (en relación a las voces de MARY) y DIF_MOD (en cuanto a las voces de Festival). También se observa que, aunque en algunos casos la diferencia sea mínima, las voces creadas en base a los audios modificados obtuvieron mejores resultados que las respectivas voces basadas en los audios originales.

Un posterior análisis estadístico indicó que existen diferencias aproximadamente significativas entre el par de voces correspondiente a UNIF (t = -1.367, df = 197.784, p = 0.08659), pero que no existen diferencias significativas para los pares de voces DIF y HMM (respectivamente: t = -0.8873, df = 194.906, p = 0.188; W = 4618.5, p = 0.1759). Estas diferencias, a pesar de no ser significativas, son consistentes en los tres tipos de voces considerados, lo cual sugiere que trabajando sobre las características de los audios base, se puede lograr un incremento en la calidad de la voz resultante. Las voces de MARY quedaron fuera de esta comparación, ya que ambos se crearon directamente a partir de los audios modificados. Para realizar el análisis estadístico se utilizaron dos métodos: ttest en todos los casos en donde se cumplieran las premisas, o wilcox en caso contrario.

[1] $z = (x - mean)/stdev$, donde $mean$ y $stdev$ son la media y la desviación estándar del participante, respectivamente.

Los tests estadísticos se llevaron a cabo ttest en los casos en donde se cumplían las premisas, o con Wilcox en caso contrario.

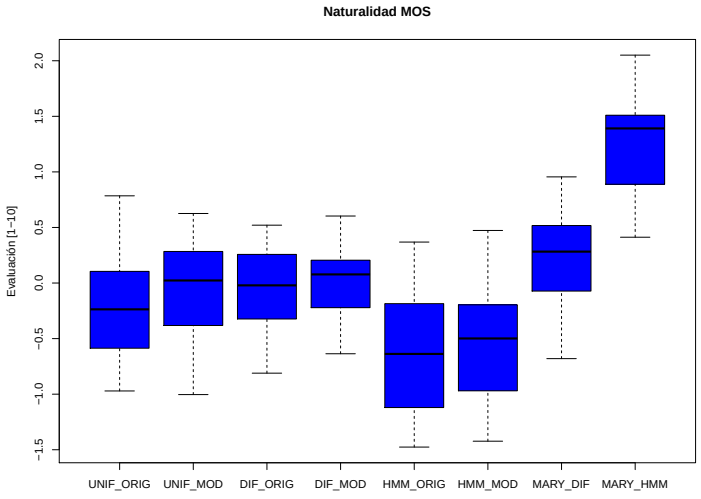

Figura 5.1: *Evaluaciones MOS normalizadas para cada voz evaluada sobre 20 participantes*

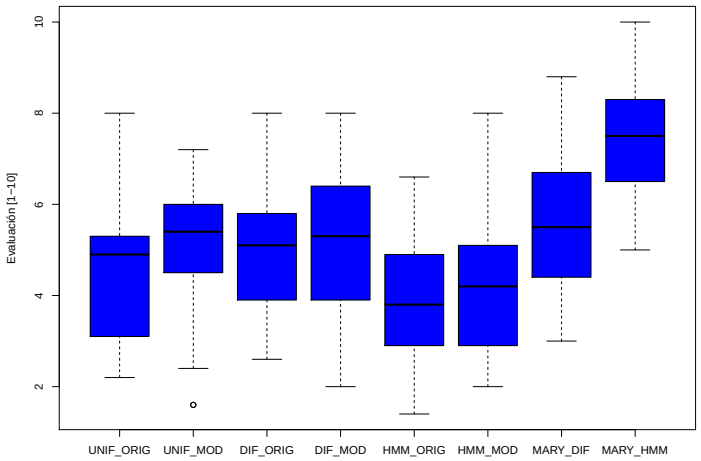

Figura 5.2: *Evaluaciones MOS para cada voz evaluada sobre 20 participantes*

Un motivo por el cual las diferencias entre los sistemas ORIG y MOD no fueron concluyentes podría ser la gran variabilidad a la hora de puntuar los audios, debido a la utilización de una escala tan amplia (1-10). Para intentar mitigar esa posibilidad, se realizó otra instancia de evaluación comparando directamente las versiones ORIG y MOD para las voces DIF y HMM de cada plataforma. En este test, al que llamaremos *choice*, participaron 10 oyentes con las mismas características que en los tests previos. Se realizó en base a las oraciones sintetizadas utilizadas en MOS, y las instrucciones dadas a cada participante fueron:

> Se le presentarán 10 pares de audios. Indique en cada caso, qué audio considera que suena más natural ("A" o "B"), o marque la opción "No noto diferencia" si opina que ambos audios suenan igual de naturales. Podrá escuchar cada par de audios la cantidad de veces que lo considere necesario.

La tabla 5.2 muestra la cantidad de votos obtenidos en el test choice para cada voz; IGUALES_DIF e IGUALES_HMM son las elecciones asociadas a la opción "No noto diferencia" de cada caso.

Elección	DIF_ORIG	IGUALES_DIF	DIF_MOD
Cantidad	21	3	26
Elección	HMM_ORIG	IGUALES_HMM	HMM_MOD
Cantidad	12	19	19

Tabla 5.2: *Resultados del test choice*

El análisis estadístico correspondiente al test choice, reflejó que no existen diferencias significativas entre las dos versiones de las voces DIF y HMM, aunque en el caso de HMM la diferencia es aproximadamente significativa (respectivamente: $t = 0.6098$, $df = 9$, $p = 0.2785$; $V = 27$, $p = 0.1108$). De hecho, en la tabla 5.2 se observa que la diferencia entre la cantidad de personas que optaron por los sistemas MOD frente a los sistemas ORIG, es baja. De todas formas, a partir de este punto analizaremos sólo los sistemas MOD, que se considera que tienden a ser mejores (o al menos no peores) que los ORIG.

La figura 5.1 sugiere que el ranking de mejor voz (MOD) construida con Festival TTS, donde "mejor" es percibido por los participantes como "más natural que", es DIF_MOD, seguido de UNIF_MOD y HMM_MOD en el último lugar. Estadísticamente, UNIF_MOD y DIF_MOD son significativamente mejores que HMM_MOD (respectivamente: $W = 6618.5$, $p \approx 0$; $W = 7238.5$, $p \approx 0$). Además, no existen diferencias significativas entre las voces UNIF_MOD y DIF_MOD (t = -0.8071, df = 189.592, $p = 0.2103$), por lo tanto no se puede afirmar que DIF_MOD sea el mejor en este caso.

En cuanto a las voces construidas con MARY TTS, no quedan dudas respecto a la superioridad de MARY_HMM sobre el resto de las voces, inclusive sobre MARY_DIF ($W = 1892.5$, $p \approx 0$).

Por último, en la figura 5.1 vemos que las evaluaciones de las voces DIF y HMM construidas a partir de MARY TTS están por encima de las de sus respectivas versiones en Festival. El análisis estadístico correspondiente, confirmó que MARY_HMM es significativamente mejor que HMM_MOD ($W = 2$, $p \approx 0$), mientras que sólo existen diferencias aproximadamente significativas entre las voces DIF de ambas plataformas (t = -1.4091, df = 33.324, $p = 0.08403$). Por lo tanto, se puede afirmar que las voces DIF y HMM construidas con MARY TTS son mejores que los construidas con Festival TTS.

5.4.2. Test de inteligibilidad SUS

Los resultados del test SUS muestran que todas las voces presentaron un alto grado de inteligibilidad, independientemente de la condición de los audios con las que fueron creadas. En la tabla 5.3 se observan los porcentajes de palabras y oraciones acertadas para cada una de las voces. La única voz MOD que exhibió diferencias significativas, en cuanto al porcentaje de palabras acertadas, frente a su respectiva voz ORIG, fue HMM ($W = 805.5$, $p \approx 0$).

A pesar de que DIF_MOD presenta mayor media de palabras correctas frente a las otras dos voces MOD de Festival, como se observa en la figura 5.3, no se puede inferir que sea la mejor voz construida bajo esa plataforma, donde ahora "mejor" es entendido como "más claro/más inteligible". Un test estadístico de los resultados, arrojó que DIF_MOD es significativamente mejor que UNIF_MOD ($W = 1080$, $p = 0.02276$), pero no lo es frente a HMM_MOD ($W = 1321$, $p = 0.1638$). Las voces construidas en base a MARY, tampoco difieren significativamente entre sí ($W = 1297$, $p = 0.269$),

por lo tanto no se puede concluir que una sea mejor que el otra.

Voz	% Oraciones correctas	% Palabras correctas
HMM_ORIG	40	92,39
UNIF_MOD	75	96,51
UNIF_ORIG	77,5	97,49
MARY_HMM	85	97,87
HMM_MOD	85	97,88
DIF_ORIG	90	98,13
MARY_DIF	90	98,38
DIF_MOD	92,5	98,89

Tabla 5.3: *Resultados de inteligibilidad SUS para cada voz, ordenados de menor a mayor según porcentaje de aciertos*

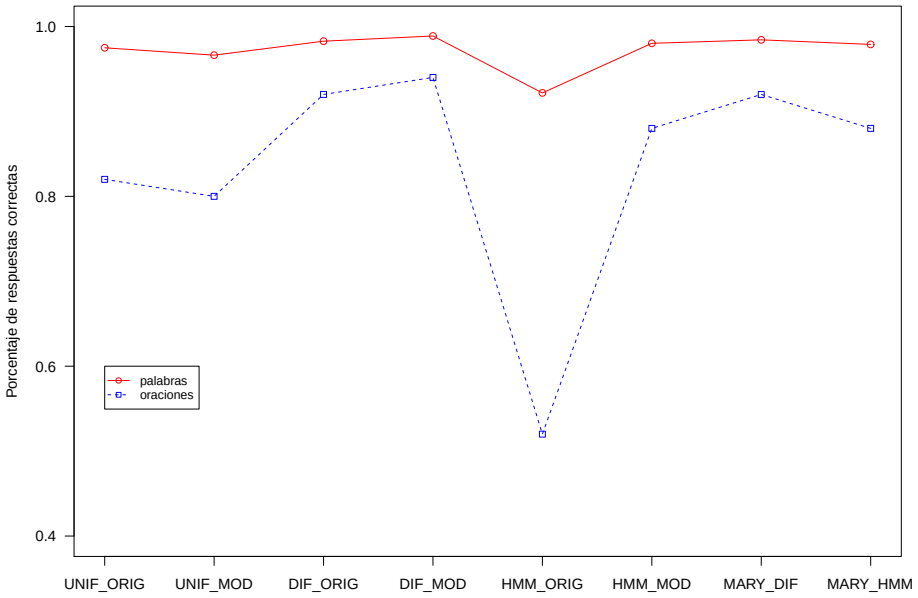

Figura 5.3: *Porcentaje promedio de palabras y oraciones correctas por frase del test SUS*

Comparamos ahora la inteligibilidad de las voces MOD (DIF y HMM) de Festival y MARY. No se puede concluir que MARY TTS sea más inteligible que Festival o viceversa, dado que los resultados no difieren significativamente en ninguno de los casos.

ORIGINAL	TRANSCRIPCION
amargo	
amargo	armado
botella	"botega"
chancho	canto
chillón	"clión"
chillón	crirón
con aire	con naile
del	de
ganaron	
la sal endulzaba la puerta	las ave vulsaba la puerta
llave	"gabe"
llave	"gaba"
nieve	
pecas	petas
perchero	
perchero	perquero
piraña	"plegarias"
quedaba	quemaba
rugosa	dudosa
rulo celeste	rulos celeste
rulo celeste	rubio celeste
sueños	suelo

Tabla 5.4: *Palabras erradas en evaluación SUS, sintetizadas con la voz HMM_ORIG. Los casilleros vacíos indican que la palabra en cuestión no fue transcripta; las palabras entre comillas fueron escritas de esa forma por los participantes.*

Para concluir, la figura 5.3 muestra también la estrecha relación entre el porcentaje de palabras correctas y el porcentaje de oraciones correctas, lo

que indica que la cantidad de errores por oración es baja. Todas las voces, exceptuando HMM_ORIG, obtuvieron buenos resultados en cuanto a la cantidad de oraciones acertadas. Además, la voz HMM_ORIG es la única, en este sentido, que difiere significativamente de su versión análoga construida a partir de los audios modificados ($W = 12.5$, $p = 0.001886$). Se puede observar, mediante la tabla 5.4, que las palabras sintetizadas con HMM_ORIG que han sido mal transcriptas por los participantes, corresponden a algunos fonemas particularmente difíciles de sintetizar y fácilmente confundibles con otros: /H/, /L/, /J/, /R/.

5.5. Resumen de los resultados

Teniendo en cuenta los resultados obtenidos en los tests y el análisis realizado en la sección anterior, ya estamos en condiciones de contestar puntualmente a las preguntas planteadas con anterioridad.

P1. ¿Cuál es la mejor voz?

DIF_MOD es la voz más inteligible según SUS, y MARY_HMM es la voz más natural según MOS. No obstante, dado que la performance en los tests SUS fue pareja entre todas las voces, se puede concluir que la voz que presenta mayor calidad general es MARY_HMM.

P2. ¿Las voces MOD son mejores que las voces ORIG? ¿En qué casos?

Las voces MOD fueron mejores que las voces ORIG en todos los casos, aunque en la mayoría de ellos la diferencia no resultó estadísticamente significativa.

P3. ¿Cuál de las tres voces construidas en base a Festival es mejor (UNIF, DIF o HMM)?

La mejor voz construida con Festival TTS es DIF_MOD.

P4. ¿Cuál de las dos voces de MARY TTS es mejor (DIF o HMM)?

La mejor voz construida a partir de MARY TTS es HMM.

P5. ¿Cuál de las dos plataformas, MARY TTS o Festival, resultó mejor para la creación de cada tipo de voz?

MARY TTS fue, sin dudas, la plataforma que creó las voces de mejor calidad general en todos los casos.

5.6. Comparación con los resultados de Gurlekian et al. 2012

En el trabajo realizado por Gurlekian et al. 2012, se comparan tres sistemas: uno basado en concatenación de unidades, desarrollado por los autores de dicho trabajo, y otros dos sistemas comerciales. Como se mencionó anteriormente, nuestros experimentos se basaron fuertemente en los experimentos llevados a cabo en [10]. Se utilizaron los mismos estímulos, las mismas instrucciones y se tuvieron en cuenta los mismos cuidados al elegir a los participantes. Por ende, estamos en condiciones de poder realizar una comparación entre los resultados obtenidos en los dos trabajos, obteniendo así una referencia de la performance lograda por las voces desarrolladas en este caso.

Con respecto al test de inteligibilidad SUS, en ambos trabajos, todos las voces evaluadas lograron un nivel de acierto comprendido entre el 90 % y 100 %. En relación al test de naturalidad MOS, dos de las voces evaluadas en el trabajo de Gurlekian et al. 2012 alcanzaron una media alrededor de 7 puntos, entre los cuales se encuentra la voz construida por ellos. En nuestro caso, la mejor voz logró una media de 7,34 puntos. Cabe recordar que esta voz fue construida mediante síntesis paramétrica, mientras que las voces evaluadas en [10], se construyeron mediante concatenación de unidades. Nuestra mejor voz basada en concatenación de unidades obtuvo 5,55 puntos. Estos resultados indicarían que, aunque aún quede mucho trabajo por realizar para conseguir una calidad óptima en los audios sintetizados, nuestras voces no se encuentran tan lejos de equipararse en calidad a otros sintetizadores existentes.

Capítulo 6

Habla expresiva

Uno de los principales objetivos al construir un sistema de habla artificial es que suene lo más natural y real posible, para que la interacción humana con estos sistemas sea agradable. Por ello, es deseable obtener habla expresiva, que sea capaz de formular preguntas, oraciones declarativas y transmitir sentimientos (como tristeza, felicidad, preocupación, enojo, entre otros).

No es una tarea para nada trivial. Como se mencionó en otras ocasiones, una de las formas de abordar esto y obtener variabilidad prosódica, es grabando una base de datos inicial en donde se plasmen diferentes emociones para cada una de las oraciones, aquellas emociones que se esperen puedan ser sintetizadas. Luego, por medio de un lenguaje con anotaciones prosódicas, como SABLE[1], podremos indicarle al sintetizador qué unidades debe priorizar y seleccionar. Otra alternativa, aunque no sencilla, consiste en modificar la señal del audio sintetizado (tono, duración, intensidad) para intentar obtener el resultado esperado.

6.1. Lenguajes de marcado

Un lenguaje de marcado (*markup language*) es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación. SABLE es un lenguaje de marcado XML utilizado para anotar textos de sistemas TTS. Sus etiquetas permiten especificar varias propiedades

[1] http://clas.mq.edu.au/synthesis/sable/sable.html

del audio a sintetizar. Definen el modo en que las palabras, números y oraciones serán reproducidas por la computadora.

Cabe aclarar, que los lenguajes de marcado no hacen más que especificar las propiedades del audio a sintetizar. Posteriormente, el *back-end* es el encargado de elegir las unidades para intentar respetar esa especificación.

Algunas de las etiquetas soportadas por SABLE son:

```
EMPH   {LEVEL = Strong, Moderate, None, Reduced}
       Establece el énfasis del texto seleccionado
BREAK  {LEVEL = Large, Medium, Small, None}
       Establece pausas prosódicas, intraoracionales
PITCH  {BASE/MIDDLE = highest, high, medium, low, lowest, deafult
       RANGE = largest, large, medium, small, smallest, default}
       Establece propiedades asociadas al pitch del texto delimitado por
       las etiquetas
RATE   {SPEED = fastest, fast, medium, slow, slowest}
       Establece la velocidad de habla del texto seleccionado
VOLUME {LEVEL = loudest, loud, medium, quiet}
       Establece el volumen del texto seleccionado
PRON   Sustituye la pronunciación dada, por la que normalmente se
       computaría para el texto seleccionado
SAYAS  {MODE = literal, date, time, phone, name, ...}
       Define el modo en que deberá decirse el texto seleccionado
```

Tanto Festival como MARY permiten la utilización de SABLE. Sin embargo, en Festival la mayoría de las etiquetas no funcionan correctamente o no son tenidas en cuenta, las únicas que funcionaron con las voces generadas fueron BREAK y VOLUME. En cambio, en MARY, el uso de casi todas las etiquetas obtuvieron el resultado esperado para la voz MARY_HMM. Lamentablemente, no fue posible su utilización en la voz MARY_DIF. Se intentó descubrir la causa por la cual las etiquetas SABLE en MARY TTS funcionaron con un tipo de voz y no con el otro, pero por falta de documentación no fue posible encontrar una respuesta.

6.2. Procesamiento de señales para generar habla expresiva

Se intentó abordar el tema desde el procesamiento de señales, ya que utilizando lenguajes de marcado no se logró. Una vez más, se utilizó la he-

rramienta Praat para esta tarea.

En este trabajo nos centramos en tres tipos de oraciones: preguntas, oraciones declarativas y oraciones sostenidas, y dos emociones: tristeza y felicidad. El primer paso fue buscar atributos acústico-prosódicos que caracterizaran dichos fenómenos, para tenerlos como referencia a la hora de modificar las señales. Los mismos se muestran en la tabla 6.1.

	Características
Tristeza	Tono bajo, mayor duración
Felicidad	Tono alto, menor duración
Pregunta	Entonación final ascendente
Oración declarativa	Entonación final descendente
Oración sostenida	Entonación final sostenida

Tabla 6.1: *Características acústico-prosódicas para habla expresiva*

Por cada emoción y tipo de oración en esta tabla, se creó un script en Praat para modificar cualquier señal de audio de la forma deseada, basándonos en pautas generales provenientes de estudios previos en los idiomas español e inglés [8] [19]. Para lograr habla triste se disminuyó el tono de todo el audio y se incrementó la duración del mismo. Para transmitir felicidad, se aumentó el tono y se decrementó la duración del audio. Para obtener entonación de pregunta, se modificó la pendiente final del tono a partir de los últimos 300 ms del audio, sin modificar la señal anterior a este punto. Se utilizó una función lineal, en donde el valor del *pitch mark* actual es el resultado de sumarle una constante al *pitch mark* anterior. En cambio, el valor en el punto de corte (duración total - 300 ms), se calcula promediando el *pitch mark* actual y sus 9 antecesores. Esto se realiza para que, en caso de que el *pitch mark* anterior al punto de corte sea un punto aislado (ruido), no altere demasiado (y de forma no deseada) la señal resultante. La misma función se utilizó para las oraciones declarativas con la única diferencia de que, en lugar de incrementar la constante, se decrementó para obtener una pendiente descendente. Por último, para lograr una oración sostenida (que dé la sensación de que la frase continúa), se realizó el mismo promedio en el punto de corte y luego se sostuvo ese valor hasta el final.

Finalmente, se creó un script en el lenguaje Python[2], que toma como input un texto, un sistema TTS (cualquiera de las voces creadas con Festival TTS o MARY TTS), y el tipo de habla deseado para la síntesis. No necesariamente el tipo de habla ingresado debe ser alguno de los presentados en la tabla 6.1, sino que podría ser cualquier combinación de ellos. Por ejemplo, se podría indicar que el resultado sea una pregunta triste.

Los resultados obtenidos fueron evaluados sólo en forma subjetiva por la autora y el director de la presente tesis, observando que los cambios realizados para obtener tristeza, preguntas, oraciones declarativas y sostenidas, parecen ser acertados. No obstante, creemos que no se logró el objetivo modificando los atributos acústico-prosódicos considerados para obtener una voz alegre, ya que las modificaciones realizadas no lograron replicar la emoción buscada en diferentes oraciones.

Nuevamente, lograr variaciones prosódicas procesando señales no es trivial. Probablemente, la tarea más compleja sea identificar correctamente qué modificar para obtener los resultados esperados, y encontrar el balance justo para lograr que suene lo más natural posible sin distorsionar demasiado la señal. Consideramos que esta técnica es una buena opción si se logra utilizar correctamente.

[2] http://www.python.org/

Capítulo 7

Conclusiones

7.1. Balance del trabajo

En este trabajo se crearon ocho voces: cinco basadas en concatenación de unidades, de las cuales cuatro fueron construidas utilizando Festival (UNIF_ORIG, UNIF_MOD, DIF_ORIG, DIF_MOD) y una a partir de MARY TTS (MARY_DIF); y tres voces basadas en síntesis paramétrica, dos a partir de Festival (HMM_ORIG, HMM_MOD) y una de MARY (MARY_HMM). Las bases de datos de unidades de los sistemas ORIG, se crearon a partir del corpus de datos SECYT. Se detectó que las variaciones de entonación que la locutora realizó en las grabaciones de los audios, producía efectos no deseables en la síntesis. Decidimos, entonces, atenuar los picos tonales para reducir la variabilidad prosódica. Para ello, modificamos el contorno de F0 de los audios mediante una función cuadrática. Con estos audios se crearon las bases de datos de las voces MOD.

Medimos la calidad de las voces construidas mediante dos tests de percepción: test de naturalidad (MOS) y test de inteligibilidad (SUS). Los resultados arrojaron que el sistema que presenta mayor calidad general es MARY_HMM. MARY fue, además, la plataforma con la que se obtuvieron mejores voces en relación a DIF y HMM. Por otro lado, la mejor voz creada a partir de Festival TTS resultó ser DIF_MOD. Lamentablemente, tanto MARY_HMM como DIF_MOD tienen problemas. DIF_MOD tiene el inconveniente de los difonos faltantes, por el cual ciertas palabras no pueden ser sintetizadas. Los problemas con las voces de MARY, en cambio, se deben a la inestabilidad de la plataforma bajo la cual están construidas.

Por último, experimentamos con modificaciones acústicas a la salida del sistema para obtener variabilidad prosódica en los audios sintetizados. Si bien el sistema construido para obtener habla expresiva no fue evaluado formalmente en este trabajo, consideramos que los audios sintetizados logran replicar de manera bastante aproximada las emociones y expresiones deseadas, salvo para el caso de una voz alegre.

7.2. Trabajo futuro

Sin duda la reducción de picos tonales utilizando una función cuadrática implicó mejoras en la síntesis y fue una buena solución al problema que introdujeron los picos tonales. Sería interesante probar con otras funciones que logren nivelar la entonación intentando distorsionar lo menos posible los segmentos de audio que no presenten picos; y evaluar nuevamente los sistemas comparando la calidad obtenida en ambos casos.

Por otro lado, nos gustaría realizar más evaluaciones sobre los sistemas construidos. Sospechamos que el test MOS, de la manera en que fue utilizado, no es el más apropiado para medir la naturalidad de los sistemas. Se empleó una escala muy grande (del 1 al 10), y observamos que cada participante sólo utilizó una pequeña porción de la misma. En parte es un fenómeno razonable, dado que la elección depende de la subjetividad del participante y de su noción de naturalidad. No obstante, restringir la escala (por ejemplo del 1 al 5) podría facilitarle al participante su decisión, logrando a su vez minimizar la dispersión de los datos que introducía la otra escala.

Por último, queda pendiente la reconstrucción de las voces bajo la plataforma MARY TTS, cuando salga una versión más estable de la misma; así como también la evaluación del sistema construido para generar habla expresiva a partir de audio ya sintetizado.

Anexos

Anexo A

A continuación mostramos el inventario fonético (en formato Scheme) utilizado en Festival para la construcción de voces del lenguaje spanish_arg (ver Subsección 3.1.1). Para construir las voces en MARY TTS se utilizó el mismo inventario fonético, pero con otro formato.

```
(defPhoneSet uba_spanish_arg
  ;;;  Phone Features
  (;; vowel or consonant
   (vc + -)
   ;; vowel length: short long dipthong schwa
   (vlng s l d a 0)
   ;; vowel height: high mid low
   (vheight 1 2 3 -)
   ;; vowel frontness: front mid back
   (vfront 1 2 3 -)
   ;; lip rounding
   (vrnd + -)
   ;; consonant type: stop fricative affricative nasal liquid
   (ctype s f a n l 0)
   ;; place of articulation: labial alveolar palatal labio-dental
   ;;                         dental velar
   (cplace l a p b d v 0)
   ;; consonant voicing
   (cvox + -)
  )
```

```
  ((#  - 0 - - - 0 0 -)

;;Vocales
   (a  + l 3 1 - 0 0 -)
   (e  + l 2 1 - 0 0 -)
   (i  + l 1 1 - 0 0 -)
   (o  + l 3 3 + 0 0 -)
   (u  + l 1 3 + 0 0 -)
   (j  + d 1 1 - 0 0 -)
   (w  + d 1 3 + 0 0 -)

;;Vocales acentuadas
   (a1 + l 3 2 - 0 0 -)
   (e1 + l 2 1 - 0 0 -)
   (i1 + l 1 1 - 0 0 -)
   (o1 + l 2 3 + 0 0 -)
   (u1 + l 1 3 + 0 0 -)

;;Consonantes
   (m  - 0 - - - n l +)
   (n  - 0 - - - n a +)
   (J  - 0 - - - n p +)

   (s  - 0 - - - f a -)
   (f  - 0 - - - f b -)
   (x  - 0 - - - f v -)
   (Z  - 0 - - - f p +)

   (b  - 0 - - - s l +)
   (d  - 0 - - - s d +)
   (g  - 0 - - - s v +)
   (p  - 0 - - - s l -)
   (t  - 0 - - - s d -)
   (k  - 0 - - - s v -)

   (l  - 0 - - - l a +)
   (r  - 0 - - - l a +)
   (R  - 0 - - - l a +)
   (L  - 0 - - - l p +)
```

```
   (H  - 0 - - - a p -))
)
(PhoneSet.silences '(#))
```

Anexo B

Difonos faltantes en la base de datos creada con Festival TTS

La siguiente tabla exhibe los difonos faltantes en la base de datos creada por Festival TTS durante la construcción de las voces DIF (ver Subsección 4.1.2).

a1	**b**	**d**	**e**	**e1**	**f**	**g**	**H**	**i**	**i1**
a1_e1	b_#	d_f	e_u1	e1_a1	f_#	g_#	H_b	i_i	i1_a1
a1_i1	b_b	d_g		e1_e1	f_b	g_b	H_d	i_i1	i1_e
a1_j	b_f	d_H		e1_f	f_d	g_d	H_f	i_j	i1_e1
a1_o1	b_g	d_l		e1_i1	f_f	g_f	H_g	i_o	i1_i
a1_u1	b_H	d_L		e1_j	f_g	g_g	H_H	i_Z	i1_i1
	b_J	d_x		e1_o1	f_H	g_H	H_J		i1_j
	b_k	d_Z		e1_u1	f_J	g_j	H_k		i1_o1
	b_L				f_k	g_J	H_l		i1_R
	b_m				f_L	g_k	H_L		i1_u1
	b_n				d_m	g_L	H_m		i1_w
	b_R				f_n	g_p	H_n		i1_Z
	b_x				f_p	g_R	H_p		
	b_Z				f_R	g_s	H_r		
					f_s	g_t	H_R		
					f_x	g_x	H_s		
					f_Z	g_Z	H_t		
							H_w		
							H_x		
							H_Z		

j	**J**	**k**	**l**	**L**	**m**	**n**	**o**	**o1**	**p**
j_a1	J_#	k_b	l_r	L_#	m_d	n_J	o_J	o1_i1	p_#
j_e1	J_b	k_d		L_b	m_f	n_L	o_w	o1_J	p_b
j_f	J_d	k_f		L_d	m_g	n_r		o1_w	p_d
j_H	J_f	k_g		L_f	m_H	n_Z			p_f
j_i	J_g	k_H		L_g	m_J				p_g
j_i1	J_H	k_J		L_H	m_l				p_H
j_j	J_i	k_L		L_j	m_L				p_J
j_J	J_j	k_m		L_J	m_m				p_k
j_L	J_J	k_n		L_k	m_n				p_L
j_o1	J_k	k_p		L_l	m_R				p_m
j_p	J_l	k_R		L_L	m_s				p_p
j_R	J_L	k_x		L_m	m_t				p_R
j_u1	J_m	k_Z		L_n	m_x				p_x
j_x	J_n			L_p	m_Z				p_Z
j_Z	J_p			L_r					
	J_r			L_R					
	J_R			L_s					
	J_s			L_t					
	J_t			L_w					
	J_w			L_x					
	J_x			L_Z					
	J_Z								

r	**R**	**s**	**t**	**u**	**u1**	**w**	**x**	**Z**	#
r_J	R_#	s_r	t_b	u_j	u1_a	w_#	x_#	Z_#	#_j
r_r	R_b	s_Z	t_f	u_o	u1_a1	w_a1	x_b	Z_b	#_J
r_Z	R_d		t_H	u_u	u1_f	w_J	x_f	Z_d	#_r
	R_f		t_J	u_u1	u1_j	w_L	x_g	Z_f	#_w
	R_g		t_k	u_w	u1_J	w_R	x_H	Z_g	
	R_H		t_L		u1_L	w_u	x_j	Z_H	
	R_J		t_p		u1_o	w_u1	x_k	Z_i	
	R_k		t_R		u1_o1	w_w	x_l	Z_i1	
	R_l		t_s		u1_u	w_x	x_L	Z_j	
	R_L		t_t		u1_u1		x_m	Z_J	
	R_m		t_x		u1_w		x_n	Z_k	
	R_n		t_Z				x_p	Z_l	
	R_p						x_r	Z_L	
	R_r						x_R	Z_m	
	R_R						x_s	Z_n	
	R_s						x_t	Z_p	
	R_t						x_x	Z_r	
	R_u1						x_Z	Z_R	
	R_x							Z_s	
	R_Z							Z_t	
								Z_w	
								Z_x	
								Z_Z	

Anexo C

A continuación presentamos las funciones utilizadas durante la construcción de todas las voces de Festival, para la predicción de prosodia y la asig-

nación de duración de segmentos (ver Subsección 4.2.1).

Árbol de decisión para la predicción de acentos

```
(set! uba_spanish_arg_accent_cart_tree
 '
  ((R:SylStructure.parent.gpos is content)
   ((stress is 1)
    ((Accented))
    ((position_type is single)
     ((Accented))
     ((NONE))))
   ((NONE)))
)
```

Función para generar contorno entonacional

```
(define (uba_spanish_arg_secyt_targ_func1 utt syl)
 "(uba_spanish_arg_secyt_targ_func1 utt syl)
Simple hat accents."
  (let ((start (item.feat syl 'syllable_start))
         (end (item.feat syl 'syllable_end))
         (ulen (item.feat (utt.relation.last utt 'Segment )
        'segment_end)) nstart nend fustart fuend fuend
        fstart fend)
        (set! nstart (/ start ulen))
        (set! nend (/ end ulen))
        (set! fustart '130)
        (set! fuend   '110)
        (set! fstart  (+ (* (- fuend fustart) nstart) fustart))
        (set! fend    (+ (* (- fuend fustart) nend) fustart))

    (cond
      ((equal? (item.feat syl "R:Intonation.daughter1.name")
        "Accented")
       (list
        (list start fstart)
        (list (+ start 0.010) (+ fstart 10 ))
        (list (- end   0.010) (+ fstart 8 ))
```

```
        (list end   fend)
))
      ((not (item.next syl))
       (list
(list end fuend)))
      ((not (item.prev syl))
       (list
(list start fustart)))
      (t
       nil)))
)
```

Función para la asignación de duración de segmentos

```
(set! uba_spanish_arg_secyt::zdur_tree
 '
  ;; clause initial
  ((R:SylStructure.parent.R:Syllable.p.syl_break > 1)
   ((R:SylStructure.parent.stress is 1)
    ((1.5))
    ((1.2)))
   ;; clause final
   ((R:SylStructure.parent.syl_break > 1)
    ((R:SylStructure.parent.stress is 1)
     ((1.5))
     ((1.2)))
    ((R:SylStructure.parent.stress is 1)
     ((ph_vc is +)
      ((1.2))
      ((1.0)))
     ((1.0)))))
)
```

Anexo D

Se presentan los listados completos de las oraciones utilizadas en la creación de los tests MOS, SUS y choice, para la evaluación de las voces construidas en este trabajo. Además, incluimos las instrucciones dadas a los

participantes en cada caso (ver Sección 5.1).

Lista de oraciones - MOS test y choice test

1. De cada seis pacientes que se van a hacer un estudio, sólo atienden a dos
2. El plantel volvió a entrenar ayer en el Parque General San Martín
3. La autora convirtió el material en un éxito de taquilla global
4. Si hubo un responsable dentro del gobierno, será sancionado
5. En las próximas horas, habría más cambios en el Gabinete
6. Y la principal preocupación de los jugadores es zafar de la Promoción
7. Este fin de semana, quedó como único puntero del torneo local
8. Hay gustos, que se pagan carísimo
9. Este no es el momento adecuado para discutir
10. Si tiene pruebas, que las presente ante la Justicia
11. Los dirigentes del gremio, confían en que la Presidente los recibirá
12. Parece que sabían los movimientos de la familia
13. El sector de informática, es el nuevo generador de empleo del país
14. La propuesta es refinanciar, y así salir de la depresión económica
15. La segunda semana fue totalmente exitosa
16. Este es un partido clave en la batalla por evitar la Promoción
17. Lo que ocurrió aquí, es algo muy terrible
18. Esperamos que resulte según lo previsto
19. El resto de la escena, se completa en forma virtual
20. En los próximos años, se estima que el clima recrudecerá lentamente
21. Cuando llueve, hay que manejar con precaución
22. Los árboles, florecen en primavera
23. Los jugadores esperan ansiosos el comienzo del partido
24. Al freírse, un alimento absorbe parte del aceite
25. Hicieron un piquete en frente del Congreso, a modo de protesta
26. Los vecinos planean hacer un cacerolazo para quejarse de la inseguridad
27. Los subsidios, son los mecanismos contrarios a los impuestos

28. Con escaso viento pero mucho entusiasmo, se largó la travesía
29. El dicho, llovido sobre mojado, tiene una connotación negativa
30. Habrá castigos, para todos los alumnos que no hagan los deberes
31. Los comerciantes de la zona de Once bajaron las persianas de los negocios a modo de protesta
32. La Plaza de Mayo y Caminito fueron emblemas turísticos del siglo veinte
33. En lugar de ir a comer, utilizan su tiempo para capacitarse y aprender
34. Hace ocho años había un ingeniero cada ocho mil habitantes, hoy hay uno cada seis mil setesientos argentinos
35. El primer recital de la banda, fue un gran éxito
36. El hambre y la desnutrición es un tema preocupante y doloroso
37. Cuando suene el timbre, todos deberán entrar al aula
38. Antes de actuar, hay que pensar y medir las consecuencias
39. Después del partido, volverán al hotel de concentración para descansar
40. El español es el segundo idioma más estudiado en Francia

Lista de oraciones - SUS test

1. El viento dulce armó un libro de panqueques
2. El insecto francés conduce el elevador
3. El avión pintaba los días de melón cocido
4. El pincel construyó algunos océanos exitosos
5. A las dos se suben los meses colorados
6. Los meses cocinan zapatos de bambú
7. Podemos atrasar el camión con chocolate de acero
8. La bicicleta contiene cinco elefantes voladores
9. El cuadro es una creación con harina de caqui
10. Sin barco nos dirigimos con presión a la peluca
11. El chancho no escribe las pinturas de flan con agua
12. Estamos por comer la galaxia sin honor
13. Los militares extranjeros beben un plato de cemento
14. Las flores bebieron un ascensor divertido

15. El viento amargo armó un libro de maní
16. El caballo de detergente conduce la heladera
17. La piraña cantó con el mate de la biblioteca
18. La perra quería dominar al café del mar
19. Estudiaba el tribunal con pelos del río
20. El cantante toma el corcho con cera paraguaya
21. Las lluvias llegan a carcajadas de un mantel
22. El camión danzaba con luces de miel rugosa
23. El libro chillón cantaba crema de zapatos
24. Ellos corrían con signos y choripanes
25. Milanesas con aire jovial de anillo
26. Los salames escribían melodías sabrosas
27. Saben de su amor por el almíbar con patas
28. El amor de la fruta de zapatos y la maceta
29. Se vacunan con semillas de llave inglesa
30. La receta es con fiambre y bulones de plástico
31. Las hojas del perchero querían contar
32. Pintaban pieles con acero quirúrgico
33. El humo de la estación cantaba muy feliz
34. El micrófono de la papa quedaba en Brasil
35. La sal endulsaba la puerta de madera
36. Todos sabían que la migraña bailaba con Violeta
37. El melón no sabía la verdad del hielo
38. Cantó con la botella del tubo fluorescente
39. El oso panda estudiaba el rollo de cocina
40. La cama corría maratón con el balero
41. Tuvieron sueños de miel con madera plastificada
42. La bolsa mostraba los ojos con alquitrán
43. No querían condimentar las uvas de cemento
44. Vamos a comer empanadas de neumáticos
45. La nieve se enamoró de un enano de polenta

46. Ganaron una casa de rulo celeste
47. Exprimieron tubos de acero con azúcar
48. Mezclaron arroz crudo y alambres de papel
49. Le gustan las películas con pecas y alfajor
50. La piedra tocaba la guitarra con las ramas

Instrucciones test MOS

Se le presentarán 40 audios. Puntúe la calidad (naturalidad) de lo escuchado en base a una escala del 1 al 10, donde 1 significa: "No suena natural en lo absoluto", y 10 significa: "Suena completamente natural". Podrá escuchar cada audio la cantidad de veces que lo considere necesario.

El primer audio será una prueba para que se acostumbre al tipo de estímulos que escuchará durante el experimento. Además, le servirá para regular el volumen a su gusto.

PRUEBA :	1.□	2.□	3.□	4.□	5.□	6.□	7.□	8.□	9.□	10.□
AUDIO 1 :	1.□	2.□	3.□	4.□	5.□	6.□	7.□	8.□	9.□	10.□
AUDIO 2 :	1.□	2.□	3.□	4.□	5.□	6.□	7.□	8.□	9.□	10.□
...										
AUDIO 40:	1.□	2.□	3.□	4.□	5.□	6.□	7.□	8.□	9.□	10.□

Instrucciones test SUS

Se le presentarán 40 audios. Escriba cada palabra que escuche en cada oración. Preste atención ya que no habrá repeticiones.

El primer audio será una prueba para que se acostumbre al tipo de estímulos que escuchará durante el experimento. Además, le servirá para regular el volumen a su gusto.

PRUEBA :
AUDIO 1 :
AUDIO 2 :
...
AUDIO 40:

Instrucciones test choice

Se le presentarán 10 pares de audios. Indique en cada caso, qué audio considera que suena más natural ("A" o "B"), o marque la opción "No noto diferencia" si opina que ambos audios suenan igual de naturales.

Podrá escuchar cada par de audios la cantidad de veces que lo considere necesario.

AUDIO 1 : A.□ B.□ No noto diferencia.□
AUDIO 2 : A.□ B.□ No noto diferencia.□
AUDIO 3 : A.□ B.□ No noto diferencia.□
...
AUDIO 10: A.□ B.□ No noto diferencia.□

Referencias

[1] J. Benesty, M.M. Sondhi, and Y.A. Huang. *Springer Handbook of Speech Processing*. Springer-Verlag New York, Inc., 2007.

[2] Alan W. Black. Clustergen: A statistical parametric synthesizer using trajectory modeling. In *INTERSPEECH*, Language Technologies Institute, 2006. Carnegie Mellon University.

[3] F. Burkhardt and W.F. Sendlmeier. Verification of acoustical correlates of emotional speech using formant-synthesis. In *ISCA Tutorial and Research Workshop (ITRW) on Speech and Emotion*, 2000.

[4] R. Carlson, T. Sigvardson, and A. Sjölander. Data-driven formant synthesis. *KTH, Stockholm, Sweden, Progress Report*, 44, 2002.

[5] F.S. Cooper, A.M. Liberman, and J.M. Borst. The interconversion of audible and visible patterns as a basis for research in the perception of speech. *Proceedings of the National Academy of Sciences of the United States of America*, 37(5):318, 1951.

[6] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357–366, 1980.

[7] Lautaro Dolberg. Asignación no supervisada de entonación para un sistema de síntesis del habla. Tesis de Licenciatura, Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Mayo 2011.

[8] J.M. Garrido Almiñana. Análisis de las curvas melódicas del español en habla emotiva simulada. *Estudios de fonética experimental*, 20:205–255, 2011.

[9] J.A. Gurlekian, L. Colantoni, and H. Torres. El alfabeto fonético SAMPA y el diseño de corpora fonéticamente balanceados. *Fonoaudiológica*, 47(3):58–69, Diciembre 2001.

[10] J.A. Gurlekian, C. Cossio-Mercado, H. Torres, and M.E. Vaccari. Subjective evaluation of a high quality text-to-speech system for Argentine spanish. In *In Proceedings of Iberspeech*, Madrid, 2012.

[11] D. Hill, L. Manzara, and C. Schock. Real-time articulatory speech-synthesis-by-rules. In *Proceedings of AVIOS*, volume 95. Citeseer, 1995.

[12] E. Moulines and F. Charpentier. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech Communication*, 9 (5/6):453–467, 1990.

[13] P.W. Nye and J.H. Gaitenby. The intelligibility of synthetic monosyllabic words in short, syntactically normal sentences. *Haskins Laboratories Status Report on Speech Research, SR-37*, 38:169–190, 1974.

[14] ITU-T Recommendation P.85. Telephone transmission quality subjective opinion tests. a method for subjective performance assessment of the quality of speech voice output devices., 1994.

[15] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.

[16] M. Schröder and A. Hunecke. Creating german unit selection voices for the MARY TTS platform from the BITS Corpora. 2007.

[17] C.H. Shadle and R.I. Damper. Prospects for articulatory synthesis: A position paper. 2002.

[18] H.M. Torres and J.A. Gurlekian. Automatic determination of phrase breaks for argentine spanish. In *Proceedings of Speech Prosody 2004*. ISCA, 2004.

[19] D. Ververidis and C. Kotropoulos. Emotional speech recognition: Resources, features, and methods. *Speech communication*, 48(9):1162–1181, 2006.

[20] M. Viswanathan and M. Viswanathan. Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (mos) scale. *Computer Speech & Language*, 19(1):55–83, 2005.