

## **Capítulo VI. Pruebas y Conclusiones**

### **6.1 Evaluación del Sintetizador**

Una vez que hemos comentado sobre como se realiza el proceso de sinterización de texto a una señal de voz, definiremos las características con las cuales creamos nuestro proyecto, esto es trataremos de medir la calidad y el desempeño en cuanto a funcionamiento del sistema creado. Dentro de la investigación localizamos algunos comentarios en cuanto a la medición, en las cuales encontramos la más adecuada para ser utilizada en este trabajo, sin embargo no dejamos atrás las opciones dadas anteriormente.

Para medir la calidad del sistema, encontramos la propuesta de Alejandro Barbosa, ya que en el desarrollo de su investigación fue parte importante de la creación del proyecto anterior, realizado por Leonardo Flores, además de que sabemos que es parte importante dentro del laboratorio de la Universidad (TLATOA).

### **6.2 Aplicación de Métricas para la Evaluación**

El desempeño y funcionamiento del sistema de sinterización del habla, nos arrojará resultados de cuales fueron los errores, y aciertos obtenidos durante las pruebas, así como detectar cual será la modificación posible para futuros trabajos, esto lo da a partir de módulos en los cuales se evalúa el funcionamiento. Estas métricas sirven específicamente para detectar el buen funcionamiento del sistema dentro de una aplicación específica. [Rodríguez, 1991].

Utilizaremos un par de valoraciones que son realizadas para calificar los TTS, y las explicaremos de la siguiente manera: Métodos Subjetivos y Métodos Objetivos.

Para Alejandro Barbosa, “la diferencia entre ellos radica en los parámetros que considera pueden ser evaluados a nivel segmental, suprasegmental y general, considerando unidades lingüísticas como fonemas, palabras y frases respectivamente”

Dentro de las **Métricas Objetivas**, son medidos los aspectos físicos, así como los patrones de voz, que son el resultado del trabajo del sintetizador, se compara con los patrones naturales [Barbosa, 1997]. Esto es mediante la medida y creación de los aspectos estadísticos sobre el número de transcripciones realizadas, y se califica en cuanto a la separación de la silabas en una palabra.

Dependiendo de los módulos que componen al sistema de sinterización, TTS, se puede crear una división, las bases para la evaluación requieren del trabajo realizado internamente por el sintetizador, con esto se determinara cuales de estos módulos fusionan correctamente, así también de la misma manera para los que no lo hagan de la misma forma. Esto da como resultado la calidad y el desempeño de un módulo.

### 6.3 Problemas y Soluciones

Los problemas que se presentaron durante la creación de las mejoras para nuestro sintetizador, en su mayoría no fueron fáciles de solucionar, debido a que no corresponden al objetivo principal de esta tesis, es decir fueron problemas detectados directamente en el corpus de voz, ya que al hacer la sintetización a nivel fonema, se detecto que el corpus en su mayoría no esta debidamente etiquetado, esto es, las palabras al ser localizadas, el sintetizador las encuentra en su milisegundo inicial y en el final, las corta y en el momento de la concatenación, se escucha la palabra de una forma la cual en ocasiones no se entiende, debido a que dentro del corpus las etiquetas no están bien dadas, además de que existen frases o palabras las cuales se encuentran grabadas en castellano lo que ocasiona un error al sintetizador al construir los wavs. La solución a esto es mejorar la etiquetación de corpus, es decir, grabar bien el corpus haciendo una separación entre cada palabra, ya que al decirlo corrido algunas sílabas o fonemas se pronuncian de manera continua.

Al tratar de darle solución al problema, se mejoró la concatenación de los fonemas y se le dio una mejor forma, pero dentro de nuestro sistema existen clases nativas del compilador de C, como lo es la `ce_concatenawav.tcl`, la cual es mandada llamar por la clase principal para realizar la creación del wav, dicha clase contiene una codificación en el lenguaje TCL, el cual desconozco, pero ahí es donde se tiene que solucionar el problema de la concatenación, asignándole ciertos parámetros para que la concatenación de las palabras sea la mejor.

## 6.4 Pruebas

Una vez que se implementó el sistema, empezamos a realizar las pruebas necesarias del sistema, con diferente tipo de oraciones, dentro de las cuales podremos ingresar una sola palabra, o una oración, del tamaño que se desee.

Durante las pruebas se obtuvo un tiempo de ejecución superior al que tenía el sintetizador anterior, esto dependiendo de la máquina donde este instalado el sintetizador, y del tamaño de la palabra a sintetizar, esto es debido a la búsqueda de los mejores candidatos a concatenar, pero aun así es muy rápida la búsqueda y la generación del audio. En cuanto al tiempo anterior, no se fue posible determinar en cuanto tiempo lo resolvía, ya que el código no corría, y en el nuestro la búsqueda y la generación del audio esta entre los 2 y 3 segundos dependiendo de lo anterior mencionado.

Los tiempos de ejecución de nuestro sistema fueron tomados en diferentes máquinas con procesadores y memorias diferentes, cabe aclarar que en cuanto el procesador era menor, la velocidad disminuía, al igual que la búsqueda.

En las siguientes imágenes podremos observar como actuaba el sintetizador y como se llevaba a cabo todo el proceso de selección para la concatenación.

```
c:\ Símbolo del sistema - java unitselection
C:\tesis>java unitselection
tu palabra es del wavs\*.* /q tamaño 2
*****
hola
EL CONTEXTO --> 1 código 010
##pal :1kkkk 1
*****
Método de depuración para candidatos
counter: 1
Si no resultaron iguales los wavs
tamaño del vector DEPURADO 2

códigos que quedaron: 000 num_palabra 0
códigos que quedaron: 010 num_palabra 1
>>>tclsh80 unitselec.tcl -file C:\user\Memo\tesis\corpus\data\corpora\gama\sintesis\speechfiles\1\timo_0007.wav -begin 509 -end 728 -name 1
tu palabra es tclsh80 unitselec.tcl -file C:\user\Memo\tesis\corpus\data\corpora\gama\sintesis\speechfiles\1\timo_0007.wav -begin 509 -end 728 -name 1 C:\user\Memo\tesis\corpus\data\corpora\gama\sintesis\speechfiles\1\timo_0007.wav
1
/tesis/wavs/1.wav
aqui
```

Fig. 6.1 Descripción del proceso de sintetización

Como se muestra en la fig. 6.1, una vez que es ingresada la palabra hola, se empieza a crear el proceso de síntesis, como solo es una sola palabra, la encuentra y le asigna un código que es el “010”, esto es por que solo encontró una palabra, y después se crea la ordena para crear el wav, como se muestra, contiene la ruta del wav así como su inicio y su fin, así como los fonemas a concatenar una vez que se haya realizado la búsqueda.

## 6.5 Conclusiones

La tesis consistió en tres partes fundamentales, la primera de ellas fue el empezar analizar el problema que se tenía, y buscar y analizar las opciones que teníamos para resolverlo. En el contenido de este documento se encuentran los tipos de síntesis mas comunes y la forma en que trabajan, como lo vimos en el primer capitulo.

Una vez que se estudio el problema y las posibles soluciones, se termina la primera fase de la tesis, en la segunda parte, la cual consistió en el diseño e implementación del sistema, así como la modificación de las clases y algunos métodos, esta parte fue, a mi consideración la mas complicada, ya que la generación de la estructura de búsqueda y las modificaciones a las clases y métodos, fueron de elaboración propia.

Como conclusiones generales, puedo comentar que estoy satisfecho con el trabajo realizado, sinceramente si nos costo un poco, en cuanto a la implementación y a la creación de las estructuras de búsqueda. Pero al final creo que se cumplió con el objetivo de la tesis, que en un principio era mejorar la calidad del sintetizador, eso se logro gracias al corpus proporcionad, pero a la vez, por problemas del etiquetado del corpus y de la clase nativa que JAVA manda llamar en nuestra clase principal, no es posible que la concatenación se lleve a cabo correctamente para algunas ocasiones.

## 6.6 Trabajo a Futuro

Aun cuando la elaboración de este trabajo fue satisfactorio, y en general se cumplieron los objetivos propuestos para ella, existe trabajo por realizar en ella, por lo que se propone el siguiente trabajo a futuro:

- Obtener un corpus aun mas limpio que el nuestro, etiquetado con una separación de tiempo considerable para que se pueda realizar bien la concatenación en cuanto a los fonemas.
- Programar, ó actualizar la clase nativa que JAVA manda llamar desde la clase principal, que es `concat_wav.tcl` asignarle los parámetros para que se realice una buena concatenación de frases y palabras.
- Cargar diferentes corpus en al importación proporcionada, es decir agregar mas de dos corpus, y probar su eficiencia.
- Que el sintetizador reconozca cualquier campo de texto, es decir, programar una clase que conecte un editor de texto cualquiera que este sea, con el sintetizador.

## 6.7 Bibliografía

- [Barbosa, 1997] A.Barbosa, *Desarrollo de una nueva voz en el español de México para el sistema de texto a voz Festival* . Tesis de Maestría, Universidad de las Américas, México Otoño 1997.
- [Barrutia, 1982] R. Barrutia. *Fonética y Fonología Españolas*. Ed. John Wiley & Sons, Inc., 1982.
- [Beutnagel & Conkie, 1999] M. Beutnagel & A. Conkie. Interaction of units in a Unit Selection Database. NJ, USA: AT&T Labs - Research, 1999.
- [Flores, 1999] L. Flores. *Unit Selection*. Tesis de Licenciatura, Universidad de las Américas, Puebla, México Primavera 1999.
- [Hunt & Black, 1996] A. Hunt & A. Black. Unit Selection in a concatenative speech synthesis system using a large speech database. In Proceedings of ICASSP-96, Atlanta, GA, 1996.
- [Internet 1]  
<http://www.alared.com/oposiciones/lengua/TEMA06.htm>
- [Meza,1999] H. Meza, *Modelos estadísticos de Duraciones de los Fonemas en el Español Mexicano*. Tesis de Licenciatura, Universidad de las Américas, Puebla, México Primavera 1999.
- [Pellom et al., 2001] B. Pellom, *University of Colorado Dialog Systems for Travel and Navigation*, Human Language Technology Conference (HLT-2001) San Diego, March 2001



- Sun Microsystems, 2004. <http://java.sun.com>
- Maslakowski M., Butcher T., Aprendiendo MySQL en 21 días. México: Pearson Education, 2001
- MySQL Site, 2004. <http://mysql.com>
- CeballosF., Java 2 Curso de programación, México: Alfa-RaMa. 2002.