Universidad Autónoma de Madrid
Escuela Politécnica Superior

Proyecto fin de carrera

DETECCIÓN DE TÉRMINOS ORALES PARA RECUPERACIÓN DE INFORMACIÓN MULTIMEDIA Y SU APLICACIÓN A VÍDEOS DE INFORMACIÓN TURÍSTICA

Ingeniería de Telecomunicación

Pablo Martín Gila
Febrero 2012

DETECCIÓN DE TÉRMINOS ORALES PARA RECUPERACIÓN DE INFORMACIÓN MULTIMEDIA Y SU APLICACIÓN A VÍDEOS DE INFORMACIÓN TURÍSTICA

AUTOR: Pablo Martín Gila
TUTOR: Doroteo Torre Toledano

Área de Tratamiento de Voz y Señales
Dpto. de Tecnología Electrónica y Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Febrero 2012

Agradecimientos

Cuando finaliza una etapa de nuestras vidas es normal y aconsejable acordarse de todas las personas que de una u otra forma han estado a nuestro lado durante esos años. Con el proyecto fin de carrera concluye un periodo que estoy seguro que aunque pase el tiempo, siempre estará entre los mejores años de mi vida, la etapa universitaria. Por supuesto, la mayor parte de la culpa de que esto sea así la tiene toda la gente que me ha apoyado, sufrido, querido, ayudado y orientado a lo largo de estos maravillosos años.

En primer lugar, me gustaría expresar un enorme agradecimiento a mi tutor en este proyecto, Doroteo Torre Toledano. Primero, por haberme brindado la excepcional oportunidad de trabajar y colaborar con él así como con el Área de Tratamiento de Voz y Señales, pero también por haberme permitido aprender de él, no sólo académicamente sino también a nivel personal. Especialmente esa formación humana es algo que también agradezco a todo el personal docente de la Escuela.

No hay palabras suficientes para expresar lo agradecido que me siento a mis padres, Mercedes y Jesús, sin los que nada de esto habría sido posible. Gracias por concederme la posibilidad de estudiar esta carrera y por el enorme apoyo prestado desinteresadamente a lo largo de estos años. Especialmente por soportarme en los momentos más difíciles, en los que siempre han estado ahí sin poner ninguna mala cara a pesar de merecerlo en muchas ocasiones. Me gustaría mostrar un agradecimiento muy especial también a mi hermana Beatriz, que siempre tiene una broma para hacerme reír y que junto con mis padres ha compartido conmigo los mejores momentos de mi vida. Por supuesto gracias a mis abuelas, que han confiado en mí en todo momento, y a toda mi familia, todos ellos me han convertido en una persona muy feliz en estos años.

Es evidente que esta etapa universitaria no significaría nada sin mis compañeros de la carrera. Todos y cada uno de ellos han aportado en mayor o menor medida un granito de arena para hacer de estos años una enorme montaña de vivencias, experiencias, aventuras, risas y diversión. Compañeros de prácticas, de estudio, de cafetería... a todos ellos, muchas gracias. Pero no puedo dejar de nombrar a tres personas muy especiales que he conocido en este periodo. Por un lado las dos personas que han estado conmigo en la Escuela y también fuera de ella desde el primero hasta el último día de estos cerca de seis años, Dani y Sergio, gracias por ser como sois. Y por supuesto a Maya, por tener siempre una sonrisa para darme ánimos y por ser la mejor de las motivaciones para terminar este proyecto, gracias.

No quiero terminar sin mencionar a mis amigos y amigas de Sanse, con los que he vivido tantas locuras estos años que no alcanzo ni a contarlas en mi cabeza. Gracias por llenar estos años con vuestra vitalidad. Por último, y para no dejarme a nadie, a todas las personas que lean esto y se den por aludidos cuando digo:

MUCHAS GRACIAS

Pablo Martín Gila
Febrero 2012

A mis padres, Mercedes y Jesús,
por ser un ejemplo a seguir en mi vida.

Resumen

Resumen

El objetivo del presente proyecto es el análisis y puesta en marcha de un sistema de detección de términos orales con la finalidad de extraer información multimedia y aplicarlo al caso concreto de unos vídeos de información turística.

El sistema básico de búsqueda de palabras clave ha sido proporcionado por la Faculty of Information Technology de la Brno University of Technology (República Checa) y la base de datos de voz utilizada proviene del audio de los vídeos de información turística con los que trabaja el entorno del proyecto MA2VICMR de la Comunidad de Madrid.

En primer lugar, se ha realizado un pequeño estudio del estado del arte principalmente centrado en los sistemas existentes en la actualidad de localización de palabras (word spotting) y más en concreto de la tecnología con la que se trabaja en este proyecto, Spoken Term Detection (STD).

El sistema completo estudiado en este proyecto se compone de un reconocedor de audio, el detector de términos orales, un conversor de formatos, un alineador de audio y un sistema evaluador de los resultados de sistemas STD proporcionado por el National Institute of Standards and Technology (NIST). Esta memoria explica con detalle todos estos elementos y bloques junto con las entradas y salidas de cada uno de ellos.

Se han realizado una serie de experimentos para comprobar el comportamiento del sistema con distintas condiciones de los parámetros de entrada. Los resultados de estos experimentos y la evaluación de los mismos se refleja en esta memoria, explicando qué ocurre en cada situación. Por último, la memoria incluye las conclusiones derivadas del estudio del sistema completo.

Palabras Clave

Alineamiento, ATWV, audio, búsqueda, curva DET, detección de términos orales, Falsas Alarmas, HMM, información multimedia, LVCSR, MA2VICMR, modelos de relleno, NIST, palabra clave, pérdidas, score, segmentación, STD, transcripción, turismo, voz y word spotting.

Abstract

The objective of this project is to analyse and to start the engine of a Spoken Term Detection (STD) system in order to extract multimedia information and use that system for some tourist information videos.

The keyword detection engine was provided by Faculty of Information Technology of Brno University of Technology (Czech Republic) and the speech data base that is used in this project comes from audio from tourist information videos. Those videos belong to MA2VICMR project financed by Region of Madrid.

First of all, a brief study of the state of the art is done to know the existing word spotting systems, mainly. It is focused in the kind of system used in this project, Spoken Term Detection (STD) systems.

The developed system consists of a speech recogniser, a Spoken Term Detection system, a format converter block, a speech aligner and a STD evaluation system provided by National Institute of Standards and Technology (NIST). This report carefully explains all these elements and blocks as well as every input and output of them.

Several experiments have been set in order to test the STD system under different conditions of input parameters. The results of the term detections and their evaluation are written in this report too, explaining what is happening in each situation. Finally, the report shows the conclusion about the study of the whole system.

Key words

Alignment, ATWV, audio, DET curve, False Alarms, filler models, HMM, keyword, LVCSR, MA2VICMR, misses, multimedia information, NIST, score, search, segmentation, speech, spoken term detection, STD, tourism, transcription and word spotting.

Índice general

Índice de figuras

Índice de tablas

1 Introducción

1.1. Motivación

En la actualidad, en el mundo desarrollado vivimos en lo que se ha denominado la sociedad de la información. En ella, la información cobra una importancia vital en todos los aspectos llegando a ser sinónimo de poder. Debido a la evolución exponencial de las tecnologías de la información y las comunicaciones, esta sociedad ha llegado a una situación en la cual hay tantos datos que no todos son ciertos o útiles en un determinado momento. De todos es sabido que cuando encuentras una determinada documentación por ejemplo en Internet, en una gran parte de los casos ésta es errónea, incompleta o no versa exactamente sobre lo que se requería.

Debido a este hecho, el nuevo reto que se nos presenta es pasar de la sociedad de la información en la que nos encontramos actualmente a la sociedad del conocimiento, en la que cobrarían importancia no simplemente los datos por sí solos sino aquellos que son verdaderos y útiles en una determinada situación.

En este entorno que se acaba de describir tienen vital importancia los sistemas de búsqueda. Al pensar en un sistema de búsqueda, parece que lo primero en lo que se tiende a pensar es en sistemas basados en texto, ya que una de las mayores fuentes de información en la actualidad es Internet, en la que una gran parte de los datos que en ella se encuentran es texto. Sin embargo, de igual importancia es la búsqueda en voz, ya que contamos con infinidad de información proveniente de vídeos o archivos de audio.

Además, suele existir un factor común a la hora de buscar y seleccionar la información dentro de un audio. Este factor es el hecho de que la mayor parte de las búsquedas suelen estar relacionadas con un país, una ciudad, un personaje real o ficticio, un monumento, una obra de arte o, en general, con nombres propios.

Es por todo ello que este proyecto se centra en evaluar la eficacia de un sistema de búsqueda de términos orales en segmentos de audio, en concreto en el audio proveniente de vídeos de información turística localizados en el popular portal de vídeos YouTube. El sistema está especialmente pensado para poder encontrar con facilidad palabras clave del tipo nombre propio como las descritas anteriormente.

1.2. Objetivos

El principal objetivo de este proyecto es la recuperación de información multimedia aplicada a vídeos de información turística a través de un sistema de detección de términos orales o Spoken Term Detection (STD). Para ello, se ha hecho uso de un sistema desarrollado por la Faculty of Information Technology de la Brno University of Technology (República Checa) denominado LatticeSTD.

El primer objetivo parcial que se marca este proyecto para alcanzar ese objetivo final es analizar los lattices de fonemas provenientes del audio de los vídeos de información turística que se usan. Estos lattices fueron obtenidos a partir del reconocimiento de ese audio realizado por José Antonio Morejón Saravia y que forman parte de su Proyecto Fin de Carrera 'Segmentación de audio y de locutores para recuperación de información multimedia y su aplicación a vídeos de información turística' [1].

Un segundo objetivo es analizar en profundidad la herramienta LatticeSTD y comprender su funcionamiento. A continuación, se pretende probar dicha herramienta con distintos ejemplos de búsqueda sobre los lattices de los que se habló anteriormente, de manera que se puedan obtener unos resultados con los que llegar a unas conclusiones sobre el comportamiento de este sistema en el caso concreto que nos ocupa.

Tras la obtención de los resultados de las búsquedas, el siguiente objetivo es poner en marcha la herramienta de evaluación de sistemas basados en Spoken Term Detection de cara a evaluar los resultados obtenidos. Esta herramienta está desarrollada por el NIST y es de libre uso. A partir de ella se pretende obtener una puntuación objetiva de los resultados que el sistema de búsqueda ha producido.

Una vez obtenidos los resultados de la evaluación de los distintos ejemplos de búsquedas, el último objetivo parcial es realizar algunas mejoras bien en el sistema de búsqueda o bien en las propias búsquedas. Mejoras como el uso del score obtenido para realizar una criba de las palabras encontradas, o el uso de una matriz de confusión a la hora de realizar la búsqueda. Finalmente, se pretende evaluar el impacto de las mejoras implementadas.

La base de datos utilizada es un conjunto de vídeos de información turística correspondientes a varios reportajes sobre distintas ciudades y regiones de España. Estos vídeos forman parte del caso de uso establecido en el proyecto MA2VICMR de la Comunidad de Madrid que se dedica a la recuperación automática de información multimedia en la red y en el que colabora el grupo ATVS de la UAM.

1.3. Organización de la memoria

Esta memoria ha sido organizada en cinco capítulos que se resumen a continuación. Como aclaración a la nomenclatura que se ha usado, cada capítulo se ha dividido en secciones y dentro de algunas secciones existen distintos apartados.

CAPÍTULO 1: INTRODUCCIÓN

En este capítulo se realiza una breve introducción acerca de la motivación y los objetivos de este proyecto.

CAPÍTULO 2: ESTADO DEL ARTE

En este capítulo se desarrolla un amplio estudio del estado del arte de los sistemas que se han utilizado a lo largo del proyecto.

En un primer lugar se habla acerca del sonido y el procesamiento humano del habla para entender desde el principio cómo se produce y se percibe la voz. Después pasa a tratar sobre el estado del arte de los sistemas de reconocimiento de voz, sección en la que se describen los sistemas que aunque no son el núcleo de este proyecto, sí han sido los sistemas usados para dar lugar a la mayor parte de información que se utiliza en este proyecto (los lattices de fonemas reconocidos), así como para complementar la evaluación del sistema objeto de este proyecto. Para concretar más, una sección de este capítulo se dedica al reconocimiento de voz con HMMs, que es el método utilizado en este proyecto.

Entrando más en detalle, el capítulo también explica todos los distintos paradigmas de Word Spotting existentes en la actualidad y para finalizar analiza en concreto la técnica usada en este proyecto, Spoken Term Detection.

CAPÍTULO 3: DESARROLLO DEL SISTEMA

Este capítulo centra su contenido en los elementos con los que ha contado este proyecto y tras ello se desarrolla cómo todos esos elementos han sido puestos en marcha y usados para llegar a obtener el sistema completo deseado. Del mismo modo, se detallan las herramientas que se han desarrollado para finalizar el sistema completo.

CAPÍTULO 4: EXPERIMENTOS Y RESULTADOS

En este capítulo se desarrolla una explicación de todos los experimentos realizados con el sistema completo descrito en el capítulo anterior. Una vez vistos dichos experimentos se pasa a mostrar los resultados obtenidos a partir de ellos.

CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO

En este capítulo se realiza una profunda reflexión acerca de los resultados obtenidos que se han expuesto en el capítulo anterior y se llega a unas conclusiones.

Además, el capítulo habla de cuáles pueden ser las líneas que seguirá esta tecnología en el futuro, así como las posibles utilidades de este sistema y sus posibles mejoras futuras.

2
Estado del arte

2.1. Introducción

A lo largo de los años, el reconocimiento de voz ha sufrido una evolución enorme gracias al exponencial crecimiento de su uso en todo tipo de aplicaciones tanto de autenticación y seguridad como forenses y de ayuda contra el crimen. Se trata de una ciencia que se ha ido abriendo hueco a pasos cada vez más agigantados ya que ha demostrado una gran fiabilidad y versatilidad en muchos ámbitos.

La voz es uno de los más importantes aspectos humanos que se pueden medir y a su vez uno de los que durante más años lleva desarrollándose su estudio. Dado su enorme número de aplicaciones, la gran facilidad que existe en cuanto a la obtención de bases de datos de voz y la cantidad de datos en forma de audio que existen en el mundo actual (es una de las formas más frecuentes de encontrar la información), se ha convertido en uno de los rasgos con mayor relevancia. La voz es precisamente en torno a lo que gira este proyecto y es por ello por lo que se dedica gran parte de este capítulo a estudiar el sonido y la producción y percepción del habla, para entender cómo funciona la voz desde sus principios. Por el mismo motivo, en este capítulo se habla del reconocimiento de voz, que ha sido utilizado en la parte previa desde la que parte este proyecto y en uno de los subsistemas desarrollados en el mismo.

Para terminar de acercarse al tema concreto en el que se centra el proyecto, este capítulo termina focalizándose en desarrollar el reconocimiento de palabras clave en audio en general, los sistemas existentes en ese campo y, más en particular, las técnicas de Spoken Term Detection que es el tipo de sistema en el que se centra el proyecto.

2.2. Procesamiento humano del habla

El habla constituye la forma más natural de comunicación entre las personas (en la Figura 2.1 se ve un esquema del proceso de comunicación oral), de ahí el gran interés que tiene el desarrollo de sistemas informáticos capaces de procesar el habla y generarla de forma automática. El procesamiento del habla abarca un amplio abanico de métodos y técnicas que sirven tanto para comprender automáticamente los mensajes como para generar artificialmente esos mensajes.

Si bien la voz es el medio de comunicación más usual, los humanos producimos y percibimos

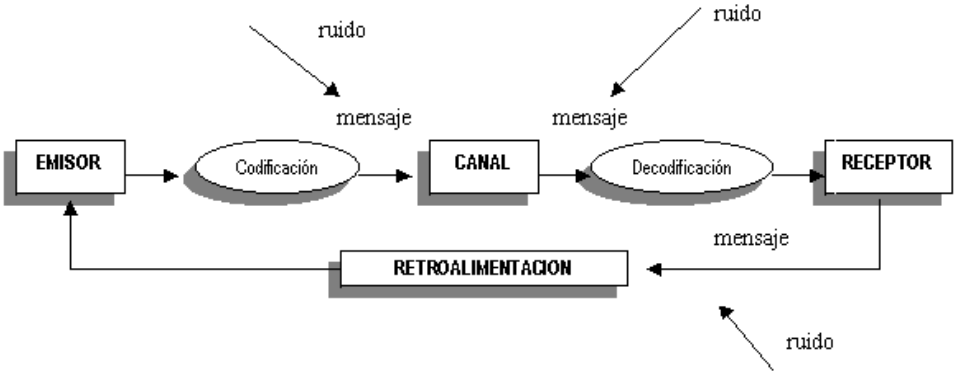

Figura 2.1: Proceso de comunicación oral

la misma con gran redundancia y de ella extraemos la información más relevante. La recuperación de información, que es el foco de este proyecto, persigue extraer los datos que el usuario desea filtrar. Por todo esto es muy importante, a la hora de realizar su tratamiento automático, determinar cómo se produce y se percibe la voz.

2.2.1. El sonido

Un sonido es una onda de presión formada por compresiones y expansiones del aire. Las compresiones son zonas donde las moléculas de aire han sido forzadas por la aplicación de energía, dando lugar a una mayor concentración de las mismas y las expansiones son zonas donde la concentración de moléculas de aire es menor.

Cuando nos referimos al sonido audible por el oído humano, lo definimos como una sensación percibida en el órgano del oído producida por la vibración que se propaga en un medio elástico en forma de ondas. El sonido audible para los seres humanos está formado por las oscilaciones de la presión del aire que el oído convierte en ondas mecánicas y finalmente, en impulsos nerviosos para que el cerebro pueda percibirlas y procesarlas. El sonido puede representarse como una suma de curvas sinusoides con un factor de amplitud diferente que pueden caracterizarse por las mismas magnitudes y unidades de medida que cualquier sinusoidal (Figura 2.2): longitud de onda (λ), período (T), frecuencia (f=1/T) y amplitud (A). También es importante la fase que representa el retardo relativo en la posición de una onda con respecto a otra.

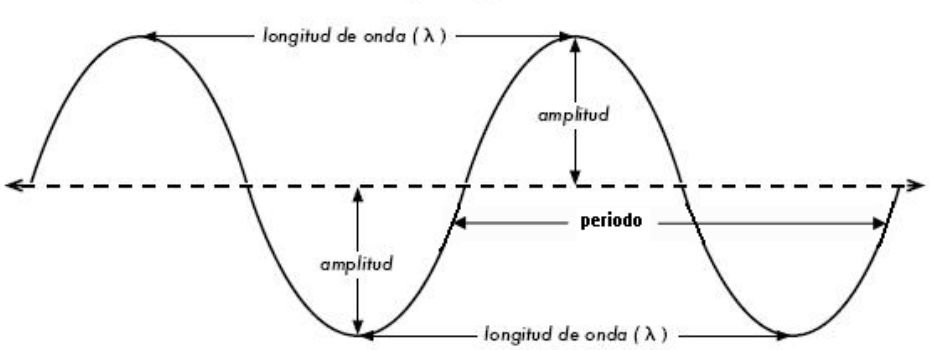

Figura 2.2: Parámetros de una onda sinusoidal

Sin embargo, un sonido complejo cualquiera no está caracterizado por los parámetros anteriores ya que, en general, un sonido cualquiera es una combinación de sinusoidales que difieren

en los cuatro parámetros anteriores. La caracterización de un sonido arbitrariamente complejo implica analizar tanto la energía transmitida como la distribución de dicha energía entre las diversas frecuencias, para ello resulta útil centrarse en:

- Potencia acústica: es la cantidad de energía por unidad de tiempo emitida por una fuente determinada en forma de ondas sonoras. La potencia acústica viene determinada por la propia amplitud de la onda, pues cuanto mayor sea la amplitud de la onda, mayor es la cantidad de energía que transmite por unidad de tiempo.

- Espectro de frecuencias: permite conocer en qué frecuencias se transmite la mayor parte de la energía.

Los sonidos de los que consta el habla se pueden clasificar por la forma en que se produce el sonido básicamente en tres tipos:

- Sonoros: Son aquellos sonidos que hacen vibrar las cuerdas vocales. Esta vibración es cuasi periódica y su espectro es muy rico en armónicos (múltiplos de la frecuencia de vibración de las cuerdas). A la frecuencia de vibración de las cuerdas se le llama frecuencia fundamental que depende de la presión ejercida al pasar el aire por las cuerdas y de la tensión de éstas. En un hombre la frecuencia fundamental se encuentra en el rango 50-250 Hz, mientras que en la mujer el rango es más amplio, encontrándose entre 100 y 500 Hz. Como ejemplos de sonidos sonoros se tiene cualquiera de las vocales (/a/, /e/, /i/, /o/, /u/).

- Fricativos: En los sonidos fricativos se produce un estrechamiento del tracto vocal por el que se hace pasar el aire, lo que proporciona como resultado una excitación de ruido aleatorio. Por ejemplo, el sonido /s/ es un sonido fricativo.

- Plosivos: Estos sonidos se producen por la existencia de una obstrucción temporal al paso del aire. El sonido se produce al abrirse la obstrucción temporal produciéndose una liberación brusca de energía en forma de una pequeña explosión. Los sonidos /p/ o /k/ son ejemplos de sonidos plosivos.

2.2.2. Producción del habla

El sistema de producción del habla no forma parte estricta del sistema sensorial humano, pero su importancia es indudable. Para determinar las operaciones de un sistema automático de reconocimiento de voz y hablante, es fundamental conocer y determinar los mecanismos que han producido un mensaje hablado, para a continuación, poder reproducirlos automáticamente. Es por ello que se van a repasar algunos conceptos fundamentales y básicos en el mecanismo de producción del habla, tanto en el órgano físico que soporta dichos mecanismos, como la producción propia del mensaje.

El habla, como señal acústica, se produce a partir de las ondas de presión que salen de la boca y las fosas nasales de un locutor. El proceso comienza con la generación de la energía suficiente (flujo de aire) en los pulmones, la modificación de ese flujo de aire en las cuerdas vocales, y su posterior perturbación por algunas constricciones y configuraciones de los órganos superiores. Así, en el proceso fonador intervienen distintos órganos a lo largo del llamado tracto vocal, que en nuestro caso asumiremos que se restringe a la zona comprendida entre las cuerdas vocales y las aberturas finales: los labios y las fosas nasales. El conjunto de órganos que intervienen en la fonación (Figura 2.3) puede dividirse en tres grupos:

- Cavidades infraglóticas (sistema sub-glotal) u órgano respiratorio: Constan de los órganos propios de la respiración (pulmones, bronquios y tráquea), que son la fuente de energía para todo el proceso de producción de voz.
- Cavidad laríngea u órgano fonador: Es la responsable de modificar el flujo de aire generado por los pulmones y convertirlo en una señal susceptible de excitar adecuadamente las posibles configuraciones de las cavidades supraglóticas.
- Cavidades supraglóticas: Están constituidas por la faringe, la cavidad nasal y la cavidad bucal. Su misión fundamental de cara a la fonación es perturbar adecuadamente el flujo de aire procedente de la laringe, para dar lugar finalmente a la señal acústica generada a la salida de la nariz y la boca.

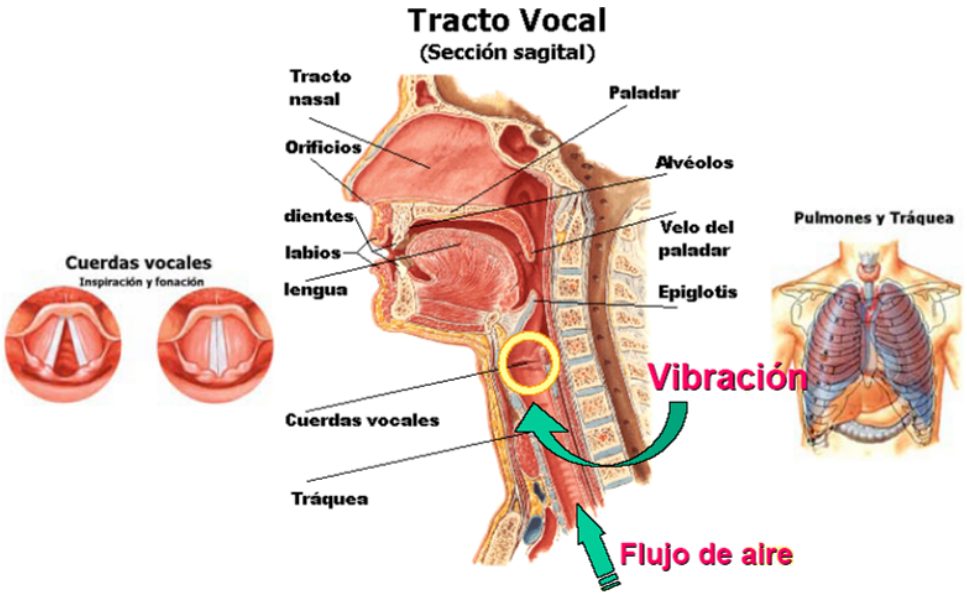

Figura 2.3: Partes del tracto vocal

2.2.3. Percepción del sonido

La capacidad de comprender el lenguaje oral se deriva del funcionamiento de un conjunto muy complejo de procesos perceptivos, cognitivos y lingüísticos que permiten al oyente recuperar el significado de un enunciado cuando lo oye.

La percepción puede verse como un proceso que une la onda acústica y su representación conceptual por medio de una serie de niveles:

- Estructura acústica.
- Estructura fonética.
- Fonología.
- Estructura superficial (información fonética).
- Sintaxis.
- Estructura profunda (información sintáctica).

- Semántica.
- Representación conceptual.

Además de las diferencias en la señal, hay también diferencias marcadas en cómo un oyente procesa los sonidos de habla y los sonidos de no habla. Para los sonidos de habla: responde ante ellos como entidades lingüísticas más que como acontecimientos auditivos. El oyente aprovecha su 'background' lingüístico para categorizar y etiquetar las señales de habla.

El problema fundamental es determinar cómo el estímulo acústico, que varía de manera contínua, se convierte en una secuencia de unidades lingüísticas discretas de forma que sea posible recuperar el mensaje. Aunque la señal de habla sea de calidad pobre o distorsionada, el proceso de percepción se realiza perfectamente. Esto se debe a que el habla es una señal altamente estructurada y redundante de modo que las distorsiones no afectan a la inteligibilidad. La percepción también es posible porque el oyente tiene dos tipos de información disponibles, el conocimiento pragmático (contexto del habla) y el conocimiento de la lengua (sintaxis, semántica y fonología).

El mecanismo físico de la percepción del habla, al igual que la audición, se realiza por medio de dos órganos fundamentales, el sistema auditivo periférico y el sistema nervioso central auditivo. El sistema auditivo periférico es lo que vulgarmente se llama oído que a su vez de divide en externo, medio e interno. En la Figura 2.4 pueden observarse las 4 partes en las que se divide el sistema auditivo: oído externo, oído medio, oído interno y el sistema nervioso central auditivo (que se sitaría tras el caracol, donde comienza el nervio auditivo).

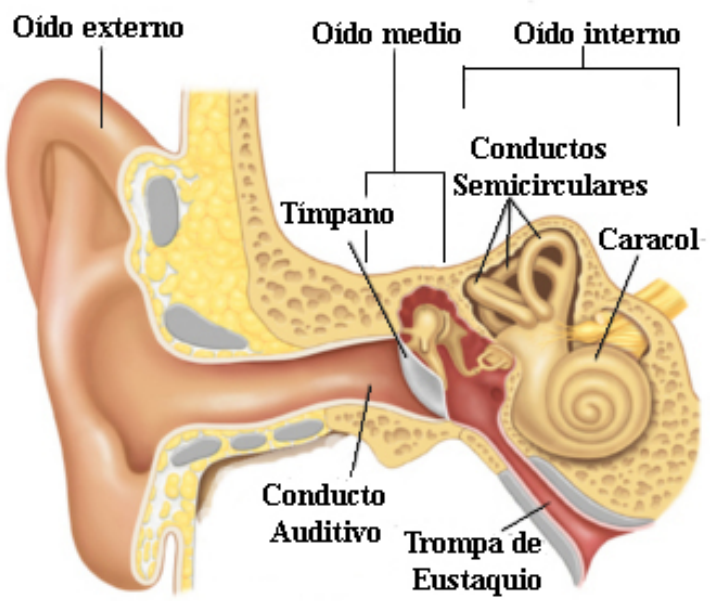

Figura 2.4: Sistema auditivo humano

El modo de funcionamiento de cada una de las partes del sistema auditivo son los siguientes:

- Oído externo: funciona por vibración del aire. Canaliza la energía acústica y está formado por la parte externa visible y el canal aditivo externo, de aproximadamente 2.5 cm, a través del cual viaja el sonido.
- Oído medio: funciona por movimiento mecánico de los huesecillos. Transforma la energía acústica en energía mecánica, transmitiéndola hasta el oído interno.

- Oído interno: primero está el funcionamiento mecánico, por el movimiento del estribo, después el funcionamiento hidrodinámico por el movimiento de los líquidos interiores de la cóclea (o caracol) y finalmente el funcionamiento electroquímico. Aquí se realiza la definitiva transformación de la energía mecánica en impulsos eléctricos.
- Sistema nervioso central auditivo: su funcionamiento es electroquímico, el movimiento de las células ciliadas provocan una reacción química que a su vez genera un impulso eléctrico.

2.3. Introducción al reconocimiento de voz

Los sonidos del habla pueden ser estudiados desde diferentes puntos de vista: articulatorio, acústico, fonético y perceptual. En esta sección se describirán desde el punto de vista fonético, acústico y articulatorio, es decir, se realizará una descripción sobre cómo se relacionan las características lingüísticas de los sonidos con posiciones y movimientos de los órganos fonatorios, así como la relación entre los fonemas y sus realizaciones acústicas, interpretando la señal de voz como la salida del proceso de producción.

2.3.1. Alófono y fonema

Antes de abordar propiamente el estudio de la fonética y la fonología, es conveniente definir primero los términos lengua y habla:

- La lengua es un modelo general y constante que existe en la conciencia de todos los miembros de una comunidad lingüística determinada, constituyendo el sistema de comunicación verbal de la misma. Es abstracto y supraindividual, es decir, está compuesto por reglas.
- El habla es la realización concreta de la lengua en un momento y lugar determinados por parte de cada uno de los miembros de esa comunidad lingüística.

El habla puede verse como una secuencia de unidades básicas, de sonidos o fonemas. Los fonemas son unidades teóricas, definidas para estudiar el nivel fonético-fonológico de una lengua humana. Son unidades lingüísticas abstractas y no pueden observarse directamente en la señal de voz. Un mismo fonema se aplica a muchos sonidos ligeramente diferentes llamados realizaciones del fonema o alófonos. Desde un punto de vista estructural, el fonema pertenece a la lengua, mientras que el alófono (sonido) pertenece al habla. La palabra 'vaso', por ejemplo, consta de cuatro fonemas (/b/, /a/, /s/, /o/). A esta misma palabra también le corresponden en el habla, acto concreto, cuatro sonidos, a los que la fonología denominara alófonos, y estos últimos pueden variar según el sujeto que lo pronuncie. La distinción fundamental de los conceptos con los que se está lidiando (fonema y alófono) está en que el fonema es una huella psíquica de la neutralización de los alófonos que se efectúan en el habla.

Los fonemas no son sonidos con entidad física, sino abstracciones mentales y formales de los sonidos del habla. Entre los criterios para decidir qué constituye o no un fonema se requiere que exista una función distintiva: son sonidos del habla que permiten distinguir palabras en una lengua. Así, los sonidos 'p' y 'd' son también fonemas del español porque existen palabras como 'pato' y 'dato' que tienen significado distinto y su pronunciación sólo difiere en esos dos sonidos. De esta forma, se puede decir que fonema es una unidad fonológica con las siguientes propiedades:

- Diferenciadora: cada fonema se delimita dentro del sistema por las cualidades que le distinguen de los demás y además es portador de una intención significativa especial.

- Indivisible: no se puede descomponer en unidades menores al contrario que por ejemplo la sílaba o el grupo fónico que sí pueden fraccionarse. Un análisis pormenorizado del fonema revela que está compuesto por un haz de diversos elementos fónicos llamados rasgos distintivos, cuya combinación forma el inventario de fonemas. El inventario de rasgos distintivos es asimismo limitado y viene a constituir una especie de tercera articulación del lenguaje.
- Abstracta: no son sonidos, sino modelos o tipos ideales de sonidos.

La distinción entre sonido y fonema ha sido un gran logro en los últimos tiempos en la lingüística. Podemos clasificar los fonemas atendiendo a dos criterios: modo de articulación y punto de articulación. En el castellano se definen 24 fonemas entre consonánticos y vocálicos. La clasificación de los primeros se observa en la Figura 2.5 según los dos criterios enunciados. Asimismo, se indica el carácter sonoro o sordo del fonema.

CLASIFICACIÓN DE LOS FONEMAS CONSONÁNTICOS DEL ESPAÑOL								
Punto de articulación		Vibración	Modo de articulación					
			Oclusivos	Fricativos	Africados	Laterales	Vibrantes	Nasales
Labiales	Bilabiales	Sordo	/p/					
		Sonoro	/b/					/m/
	Labiodentales	Sordo		/f/				
		Sonoro						
Dentales	Interdentales	Sordo		/θ/				
		Sonoro						
	Dentales	Sordo	/t/					
		Sonoro	/d/					
Alveolares		Sordo		/s/				
		Sonoro				/l/	/r/, /rr/	/n/
Palatales		Sordo			/ch/			
		Sonoro		/y/		/ll/		/ñ/
Velares		Sordo	/k/	/x/				
		Sonoro	/g/					

Figura 2.5: Clasificación de los fonemas consonánticos del español

Las vocales en español y en la mayoría de idiomas no se suelen clasificar de la manera que se ha explicado anteriormente sino que responden a una clasificación más sencilla atendiendo a la posición de la lengua (anterior, media o posterior) y a la abertura de la boca (cerradas, medio cerradas o abiertas), tal y como se ilustra en la Figura 2.6.

Posición de la lengua / Abertura de la boca	ANTERIORES	CENTRALES	POSTERIORES
CERRADAS	i		u
MEDIO CERRADAS	e		o
ABIERTAS		a	

Figura 2.6: Clasificación de los fonemas vocálicos del español

Como se ha visto, un sonido o alófono es cualquiera de las posibles realizaciones acústicas de un fonema y se caracteriza por una serie de rasgos fonéticos y articulatorios. El número de dichos rasgos y la identificación de los mismos es tarea de la fonética. En cambio, la fonología no necesariamente trata entes claramente distinguibles en términos acústicos. Como realidad mental o abstracta, un fonema no tiene por qué tener todos los rasgos fonéticos especificados.

Por ejemplo, en algunas lenguas como el chino mandarín la aspiración es relevante para distinguir pares mínimos (cada fonema tiene predefinido su grado de aspiración). Sin embargo, un fonema del español puede pronunciarse más o menos aspirado según el contexto y la variante lingüística del hablante, pero en general no está especificado el grado de aspiración.

Dada la distinción entre fonema y alófono, existe otra forma de concebir un fonema como una especificación incompleta de un conjunto de rasgos fonéticos comunes a todos los alófonos que forman la clase de equivalencia del fonema.

Fijado un conjunto de rasgos fonéticos se pueden definir los sonidos de la lengua. En principio no hay límite a lo fina que pueda ser la distinción que establecen estos rasgos. Potencialmente la lista de sonidos puede hacerse tan grande como se quiera si se incluyen cada vez más rasgos. Sin embargo, el número de fonemas es un asunto diferente, puesto que muchos de los anteriores sonidos serían equivalentes desde el punto de vista lingüístico.

2.3.2. Modelos fonéticos

Los fonemas, tal y como se ha citado en el apartado anterior, representan un nivel superior de fragmentación de palabras. La modificación de un fonema puede cambiarle el sentido a la misma. Hay que distinguirlo de alófono, que es cada una de las pronunciaciones reales del modelo ideal que representa el fonema. En función del grado de resolución que se quiera que presenten las unidades fonéticas se pueden obtener modelos más o menos específicos. La manera en que se hace la división en unidades fonéticas depende del contexto donde el alófono se localice:

- Monofonemas: son unidades totalmente libres de contexto. Un monofonema tiene en cuenta todas las posibles realizaciones de un fonema independientemente de sus vecinos.
- Bifonemas: son unidades que dependen solo de uno de sus contextos, ya sea este el derecho (bifonema derecho) o el izquierdo (bifonema izquierdo).
- Trifonemas: son unidades que dependen de ambos contextos a la vez.
- Trifonemas generalizados: Dado que el número de unidades va aumentando al ir considerando más detenidamente la posición de los fonemas en su contexto, puede llegar a ser tan elevado que su entrenamiento no fuera posible. Surge el trifonema generalizado como un primer nivel de compartición en el que varios trifonemas cercanos se agrupan para reducir el número de modelos y para que el entrenamiento de estos sea mejor.

De esta forma, un reconocedor necesita disponer de un conjunto de unidades que permita construir cualquier palabra o frase a partir de su concatenación. Los fonemas representan este conjunto completo y reducido de unidades a partir de las cuales se puede generar cualquier palabra. Estas unidades pueden modelarse con mayor o menor resolución en función del contexto a considerar.

La creación de modelos acústicos, para su posterior uso en reconocimiento de habla, se realiza en dos fases:

- Extracción de las características del sonido del habla: Por semejanza con el funcionamiento del sistema humano, la extracción de esas características, que llamaremos parámetros, se realiza en el dominio de la frecuencia. Es la asignación de los parámetros extraídos a las representaciones discretas de nuestro diseño (fonemas, trifonemas, palabras, etc) correspondientes, con el objetivo de crear un modelo para cada una de las representaciones discretas que las identifique.

- Entrenamiento y reconocimiento de modelos para cada fonema a identificar: A partir de la extracción de parámetros se construirán una serie de modelos estadísticos con los cuales se identificarán, con cierta probabilidad, fonemas en otras locuciones. Por ello, se mide la distancia entre el conjunto de parámetros que constituye el modelo y los parámetros de la pronunciación a reconocer. Existen varias técnicas para realizar el proceso:

  - HMM (Hidden Markov Model): las aproximaciones estadísticas toman como referencia el modelo estocástico de los datos. Se basa en la creación de modelos de fonemas en estados. Hasta la fecha este método es el que mejores resultados proporciona y el más utilizado. Esta fue la técnica usada en el reconocimiento de voz realizado en el PFC 'Segmentación de audio y de locutores para recuperación de información multimedia y su aplicación a videos de información turística' de José Antonio Morejón Saravia [1], voz reconocida de la cual ha partido este proyecto. Esta técnica ha sido usada también en el alineamiento de voz realizado en este proyecto. En la Figura 2.7 se puede ver un ejemplo de los estados de un HMM.

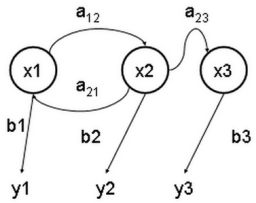

Figura 2.7: Ejemplo de el diagrama de estados de un HMM sencillo

  - DTW (Alineamiento Temporal Dinámico): consiste en alinear de forma temporal los parámetros del archivo de test y los parámetros de los modelos, obteniendo la función que alinea a ambos, eligiendo la función de menor coste posible para dicha adaptación, como se ve en la Figura 2.8.

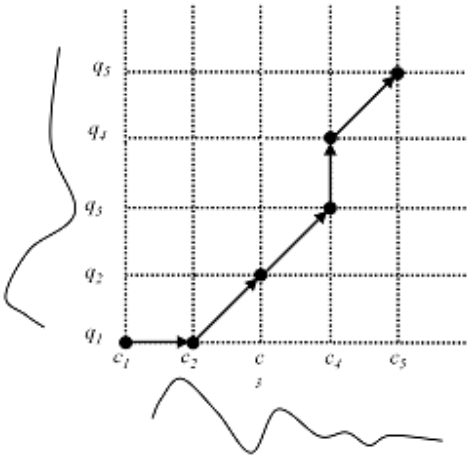

Figura 2.8: Ejemplo de alineamiento temporal dinámico

  - VQ (Cuantificación vectorial): consiste en representar las características de los fonemas como un espacio vectorial de dimensión el numero de parámetros, de forma que al fonema a reconocer se le asigna el vector cuya distancia a él sea mínima. Por tanto, los fonemas quedarán representados por unos vectores determinados (centroides) de forma que todos los puntos que caigan en una zona determinada se asignarán a dicho vector. Esto puede verse en la Figura 2.9 en la que el espacio es bidimensional (el número de parámetros que se emplea es dos).

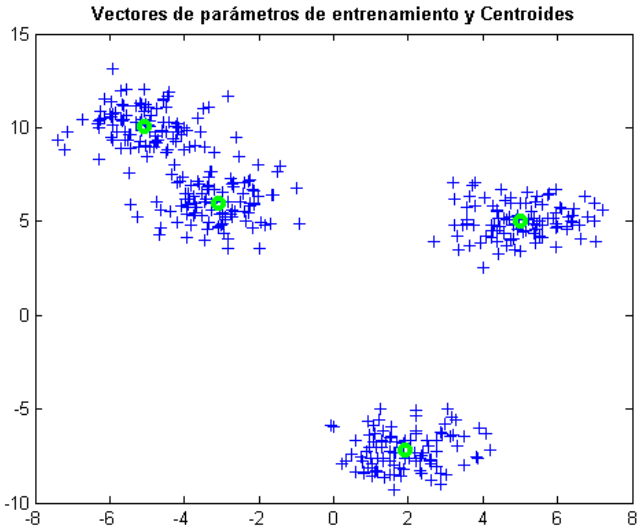

Figura 2.9: Ejemplo de cuantificación vectorial

2.4. Reconocimiento de voz con HMMs

Los Modelos Ocultos de Markov fueron descritos por primera vez por Baum [2]. Poco después, fueron aplicados al reconocimiento automático del habla en CMU [3] e IBM [4]. En los útimos años se han convertido en la aproximación predominante en reconocimiento del habla debido a la simplicidad de su estructura algorítmica y a sus buenas prestaciones. Por ello ha sido el método utilizado en el reconocimiento de voz previo al sistema del que este proyecto es objeto, así como en el alineamiento de voz realizado en este proyecto.

2.4.1. Definición y tipos de Modelos Ocultos de Markov

Un modelo oculto de Markov o Hidden Markov Model (HMM) es la representación de un proceso estocástico que consta de dos mecanismos interrelacionados: una cadena de Markov de primer orden subyacente, con un número finito de estados, y un conjunto de funciones aleatorias, cada una de las cuales asociada a un estado. En un instante discreto de tiempo se supone que el proceso está en un estado determinado y que genera una observación mediante la función aleatoria asociada. Al instante siguiente, la cadena subyacente de Markov cambia de estado siguiendo su matriz de probabilidades de transición entre estados, produciendo una nueva observación mediante la función aleatoria correspondiente. El observador externo sólo sabe la salida de las funciones aleatorias asociadas a cada estado, siendo incapaz de observar directamente la secuencia de estados de la cadena de Markov, de ahí el nombre de modelo oculto.

Entonces, un modelo oculto de Markov es la composición de dos procesos estocásticos (X,Y) definidos como:

- Una cadena oculta de Markov X que tiene en cuenta la variabilidad temporal, y que no es directamente observable.
- Un proceso observable Y que tiene en cuenta la variabilidad espectral y va tomando valores en el espacio de las características acústicas u observaciones.

La combinación de ambos procesos modela las fuentes de variabilidad de la señal de voz y permite reflejar una secuencia de parámetros acústicos como concatenación de los procesos

elementales del modelo con la flexibilidad suficiente para hacer sistemas de reconocimiento. Los Modelos Ocultos de Markov usados en el reconocimiento de voz tienen dos asunciones formales características:

- La historia de la cadena no influye en la evolución futura de la misma si existe información actual (hipótesis de Markov de primer orden).
- Ni la evolución de la cadena ni las observaciones pasadas determinan la observación actual si se ha especificado la última transición de la cadena (hipótesis de independencia de las salidas).

2.4.2. Problemas a resolver para la utilización de un HMM

Los Modelos Ocultos de Markov se caracterizan por tres problemas que hay que resolver para que resulten modelos útiles en aplicaciones reales:

- Puntuación: Dada una observación acústica y un modelo oculto de Markov, determinar la probabilidad de que el modelo genere esa observación. Esta probabilidad se calcula con el algoritmo forward-backward [5].
- Reconocimiento de estados: Dada una observación acústica y un modelo oculto de Markov, determinar de la secuencia de estados que mejor 'explica' la observación . Se lleva a cabo mediante el algoritmo de Viterbi [6].
- Entrenamiento: Dada una colección de secuencias observables, ajustar los parámetros del modelo HMM para maximizar la probabilidad de observar el conjunto de entrenamiento dado el modelo. Este problema se resuelve comúnmente con el algoritmo Baum-Welch [7].

2.5. Reconocimiento de palabras clave (Word spotting)

Dentro del mundo del reconocimiento de voz, cobra una importancia vital el reconocimiento de palabras clave o word spotting, de cara a la extracción de información de los segmentos de audio a reconocer. Esta importancia radica en el hecho de la existencia de una ingente cantidad de información de lenguaje oral en formato de audio.

Hasta hace unos años se pensaba que el problema del reconocimiento de palabras clave estaba resuelto con los sistemas de word spotting basados en reconocedores de habla continua de gran vocabulario o Large Vocabulary Continuous Speech Recognition (LVCSR). Este tipo de sistemas se basan en un diccionario que suele contener unas 65.000 palabras [8] y tienen la mejor tasa de acierto siempre y cuando las palabras que se buscan estén dentro del vocabulario. El problema de este tipo de sistemas viene cuando la palabra que se desea reconocer es una de las llamadas palabras fuera de vocabulario o Out-Of-Vocabulary (OOV), que son normalmente nombres propios, acrónimos o extranjerismos. Según un estudio realizado por Logan [9], el 12 % de las palabras que forman parte de una consulta típica de un usuario son OOV. A ello se le suma el hecho de que los idiomas van creciendo con lo que aparecen nuevas palabras (unas 20.000 cada año [10]) que no están en el vocabulario a no ser que éste se actualice.

Por todos estos hechos son necesarios sistemas que realicen un reconocimiento más eficiente de las consultas de los usuarios o que al menos sean un buen complemento de los sistemas basados en LVCSR. Es por ello que aparecen otros dos grandes tipos de word spotting: los sistemas basados en modelos de relleno y los sistemas basados en reconocedores de voz de sub-unidades de palabra.

Sistema	FOM
LVSCR	66.95
Modelos de relleno	64.46
Sub-unidades	58.90

Tabla 2.1: Tasa de aciertos de cada uno de los sistemas de word spotting

El primero de ellos, a pesar de tener la mejor tasa de aciertos por detrás de los sistemas LVCSR, tiene el problema de que en caso de cambiar alguna palabra a buscar, hay que reprocesar todo el audio.

Por el contrario, los sistemas basados en sub-unidades de palabra no necesitan reprocesar todo el audio ya que están basados en algo que no cambia dentro de un idioma: los fonemas, grafemas, sílabas, etc. La parte mala de estos sistemas es que cuentan con la peor tasa de aciertos de los tres sistemas de los que se está hablando pero son la mejor solución y complemento para paliar la principal carencia de los sistemas LVCSR: la búsqueda de palabras fuera de vocabulario. La Tabla 2.1 muestra la tasa de aciertos (FOM) de cada uno de los sistemas. En ella, FOM está mostrado en número de aciertos por cada 5 falsas alarmas por palabra clave por hora.

Por lo tanto, los dos sistemas más utilizados para el reconocimiento de palabras clave son los basados en LVCSR y los basados en sub-unidades de palabra. En la Figura 2.10 se muestra cómo estos dos sistemas se complementan para conseguir búsquedas que abarquen cualquier término que se le pueda ocurrir a un usuario.

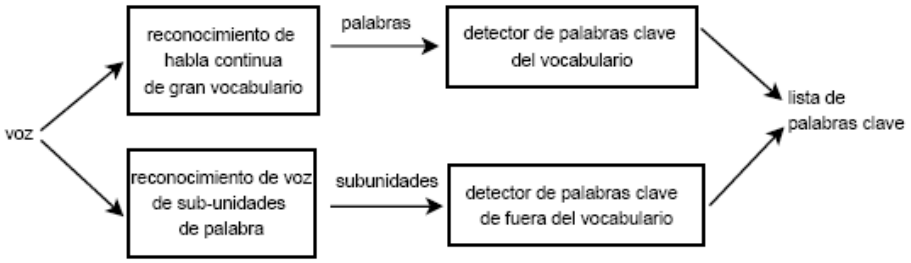

Figura 2.10: Combinación de un sistema de word spotting basado en un reconocedor LVCSR con un sistema basado en sub-unidades de palabra

En cualquiera de estos sistemas uno de los factores más importantes para verificar la calidad de los mismos es la relación entre aciertos y falsas alarmas.

2.5.1. Word spotting basado en reconocedores de habla continua de gran vocabulario (LVCSR)

Como se ha comentado anteriormente, los sistemas de word spotting basados en reconocedores de habla continua de gran vocabulario (LVCSR) utilizan un diccionario completo de la lengua concreta en la que estemos buscando términos a partir del cual se hace un reconocimiento cuya unidad más pequeña es la palabra. Tras ese reconocimiento y con él la obtención de todas las palabras presentes en el audio, se hace uso de un detector de palabras clave con el que obtenemos la localización de los términos que estamos buscando. En la Figura 2.11 se observa un diagrama de bloques de este proceso.

Una forma común de organizar los vocabularios usados para estos sistemas es lo que se llama

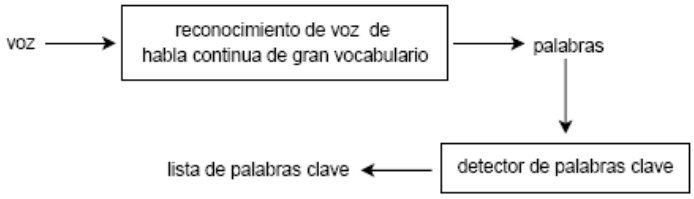

Figura 2.11: Diagrama de bloques de un sistema de word spotting basado en un reconocedor de palabras de gran vocabulario

vocabulario en árbol o 'tree lexicon', que reduce el tiempo de búsqueda aunque introduce algunos problemas prácticos a la hora de confiugurarlo [11].

El gran problema de estos sistemas es la imposibilidad que tienen de encontrar términos fuera de vocabulario (OOV), ya que en el diccionario que usan nunca están presentes nombres propios, acrónimos o extranjerismos.

A ello se le suma que estos sistemas tienen un importante problema de robustez ya que su tasa de error es elevada si las condiciones del audio y la voz sobre las que trabaja el reconocedor no coinciden con las de entrenamiento.

A pesar de ello, estos sistemas son muy útiles para la recuperación de información siendo prácticamente imprescindibles en un sistema de búsqueda de palabras clave. Sin embargo, necesitarán de un sistema complementario que les ayude con las palabras OOV.

2.5.2. Word spotting basado en modelos de relleno

Los sistemas de word spotting basados en modelos de relleno fueron propuestos por Rose y Paul [12] en el año 1990 y se basan en un diccionario que sólo contiene los términos que se desean buscar en el audio y en unos modelos de relleno.

A la salida del sistema reconocedor se obtiene el conjunto de los términos buscados así como los modelos de relleno que se sitúan en las zonas del audio donde no existen palabras clave. Tras ello, mediante unas medidas de confianza (debidamente diseñadas para descartar las hipótesis propuestas con una puntuación o score muy bajo) se obtiene la localización de los términos buscados. En la Figura 2.12 se puede observar un diagrama de bloques de este tipo de sistemas.

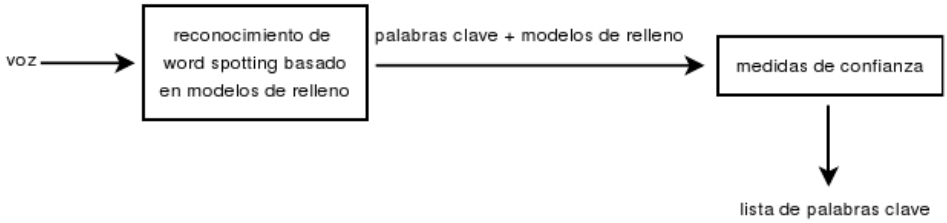

Figura 2.12: Diagrama de bloques de un sistema de word spotting basado en modelos de relleno

Existen distintas formas de entrenar tanto los modelos de relleno como los de palabras clave. Esta forma de entrenar los modelos, y sobre todo las unidades de palabra (fonemas, sílabas, palabras, etc) en las que se basa el entrenamiento, es de vital importancia. A lo largo del tiempo se han propuesto distintas formas de hacerlo:

- Rose y Paul [12] llegaron a la conclusión de que los mejores resultados se daban usando modelos basados en fonemas dependientes de contexto, usando las regiones de audio en las que no aparecen palabras clave para los modelos de relleno y todo el audio para los modelos de palabras clave. A éstos les seguían en calidad los modelos basados en fonemas independientes de contexto y los peores fueron los modelos de palabras.
- Manos y Zue [13] obtuvieron los mejores resultados para modelos de relleno usando agrupación (clustering) de fonemas independientes de contexto en clases amplias (oclusivas, nasales, fricativas, etc). Para palabras clave el mejor entrenamiento fue el realizado con regiones en las que aparecen palabras clave.
- Cuayahuitl y Serridge [14] observaron para un sistema para el idioma español, tras comparar modelos de fonema, sílaba y palabra, que el modelo de sílaba era el que mejores resultados obtenía al ser más robusto que el de fonema y cubrir todo el idioma, no como el de palabra.

También existen varias formas de realizar las medidas de confianza de las hipótesis obtenidas:

- Una de ellas es la propuesta por Cuayahuitl y Serridge [14], Xin y Wang [15], Tejedor et.al [16] o Szoke et.al [17], que consiste básicamente en descartar las hipótesis cuya puntuación (score) sea menor que un umbral.
- Tejedor et.al [18] se basaron en la semejanza entre la salida de un reconocedor de fonemas y lo que proponía el sistema basado en modelos de relleno.
- Otras propuestas han sido la de Ou et.al [19], que descartaban hipótesis según lo que determinaba una red neuronal basada en varias características (puntuación global, puntuación de las N-mejores hipótesis, etc) o la de Ben Ayed et.al [20, 21] que usaron máquinas de soporte vectorial (SVM).

2.5.3. Word spotting basado en reconocedores de voz de sub-unidades de palabra

El estudio en profundidad de los sistemas de word spotting basados en reconocedores de voz de sub-unidades de palabra ha crecido a raíz de la aparición de un tipo de sistemas enclavado dentro de esta clasificación, los sistemas de Spoken Term Detection (STD), que es el objeto principal de este proyecto. Más en concreto, la evolución de estos sistemas fue desencadenada tras la evaluación del NIST de STD en 2006 [22].

Como se puede observar en la Figura 2.13, este tipo de sistemas siempre están formados por una primera fase off-line en la que se realiza el reconocimiento de voz basado en sub-unidades de palabra (habitualmente fonemas, pero también grafemas, sílabas, grafones, etc) y una segunda fase on-line en la que se hace uso de un detector de palabras clave que se encarga de buscar las palabras en la secuencia reconocida en la primera fase. Además, se suelen realizar unas medidas de confianza para descartar falsas hipótesis. Este proyecto se centra principalmente en la segunda fase (detector de palabras clave) y en las medidas de confianza.

La utilidad de los sistemas de word spotting basados en reconocedores de voz de sub-unidades de palabra y en concreto de los sistemas de STD, radica en el hecho de que los fonemas, sílabas, etc, en definitiva, las sub-unidades de palabra se mantienen inalterables en un idioma. De esta forma, estos sistemas se convierten en idóneos para la localización de palabras fuera de vocabulario (OOV).

En los últimos años se han propuesto diversas variantes de este tipo de sistemas:

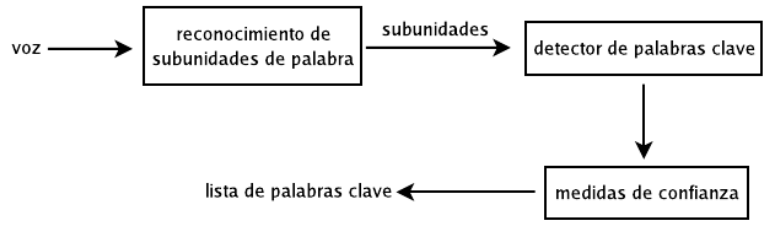

Figura 2.13: Diagrama de bloques de un sistema de word spotting basado en un reconocedor de sub-unidades de palabra

- Amir et.al [23] propusieron un sistema en el que se obtenía una secuencia de fonemas que correspondía a la secuencia de palabras más probable reconocida por un reconocedor de palabras de gran vocabulario. Es de esa secuencia de fonemas de donde se obtienen las hipótesis tras la búsqueda de términos. Este sistema se ayudaba además de una matriz de confusión en la que se encontraban las probabilidades de inserción, borrado y sustitución de cada fonema.

- Thambiratnam y Sridharan [24] y Szoke et.al [17] desarrollaron sistemas basados en un reconocedor de fonemas en los que para cada instante de tiempo existía más de una hipótesis del fonema encontrado, teniendo en cuenta también las probabilidades de borrado, inserción y sustitución. De esta forma se configuraba un grafo (lattice) sobre el que se realizaban las búsquedas que daban lugar a las hipótesis.

- También se han desarrollado sistemas que usan grafemas, sílabas o grafones (que son fragmentos de palabra que se entrenan a partir de la secuencia de fonemas y grafemas de las palabras) como sub-unidades de palabra. Tejedor et.al [25] usó grafemas demostrando que puede llegar a mejorar a los fonemas para el idioma español por la estrecha relación grafema-fonema y el menor número de los primeros. Wang et.al [26] demostró algo parecido para el idioma inglés y las palabras OOV. Akbacak et.al [27] obtuvieron muy buen rendimiento de los grafones para las palabras OOV mientras que Larsson et.al [28] propusieron un reconocedor silábico con el que configuraban un grafo similar al de los sistemas basados en reconocedor de fonemas.

Si hablamos de los sistemas de medida de confianza, es muy común el uso de redes neuronales para afinar la puntuación final de las hipótesis respecto a la que obtiene directamente el reconocimiento.

Acerca de los sistemas de Spoken Term Detection se hablará con más detalle en la sección 2.6 de este capítulo.

2.5.4. Combinación de sistemas de word spotting

De cara a obtener un mejor rendimiento de los sistemas de word spotting, a lo largo de los años se han propuesto distintas alternativas que se basan en la combinación de sistemas de los anteriormente explicados. En líneas generales, la combinación más usada debido a ser la que mejores resultados devuelve es la combinación de un sistema basado en un reconocedor de palabra de gran vocabulario y uno basado en un reconocedor de sub-unidades de palabra (normalmente de fonemas). Al elegir el momento de combinar, hay teorías en las que se combina durante el proceso de reconocimiento y otras en las que se combinan los resultados obtenidos al final de cada uno de los sistemas usados.

Sistema	ATWV
Reconocedor de palabra	72.5
Reconocedor de palabra + grafones	74.2
Sub-unidades	58.90

Tabla 2.2: Tasa de resultados de distintos sistemas

Sistema	FOM
Reconocedor de palabra	49.3
Reconocedor de fonemas	64.0
Combinación	73.9

Tabla 2.3: Tasa de resultados de los distintos reconocedores

Una de las propuestas de combinación es la publicada por Yu y Seide [29] en la que experimentaron con dos cosas. Por un lado, un reconocedor híbrido con fonemas, sílabas y palabras completas que producía un grafo en el que había sub-unidades de esos tres tipos. Por otro lado, la combinación de dos reconocedores, uno de palabras y otro de fragmentos de palabra cada uno con su correspondiente grafo (la búsqueda se realiza en ambos grafos salvo para las palabras OOV que sólo se encuentran en el de fragmentos de palabra). Ambos experimentos dan a luz resultados mejores que los de cualquiera de los reconocedores por separado, como se puede observar en la Tabla 2.2., donde cuanto más grande es ATWV, mejor es el sistema (se explicará este valor más adelante).

Iwata et.al [30] combinaron las salidas de dos sistemas de búsqueda, uno basado en palabra y otro basado en fonemas. Las hipótesis de la salida final eran aquellas que aparecían en la salida de ambos reconocedores. Esta combinación proporcionaba resultados mejores que el uso de únicamente un sistema reconocedor.

Otra de las propuestas es la realizada por Szoke et.al [31] y Vergyri et.al [32], donde se usan grafones para las palabras OOV. Por otro lado se usa un reconocedor basado en palabras para los términos dentro de vocabulario con el que se configura el correspondiente lattice de palabras. Los resultados son mejores que usando únicamente un reconocedor de palabra, como se observa en la Tabla 2.3.

Miller et.al [33] propusieron tener dos lattices: uno de palabras y otro de fonemas, construido a partir de la transcripción fonética de las palabras que hay dentro del lattice de palabras. Así, a las palabras dentro del diccionario se accede a través del primero de los lattices mientras que a las OOV se accede a través del segundo, con su correspondiente matriz de confusión.

2.6. Detección de términos orales (Spoken Term Detection)

2.6.1. Características de STD

Dentro de los sistemas de reconocimiento de palabras clave o word spotting se pueden considerar como un caso especial los sistemas de detección de términos orales o Spoken Term Detection (STD) dado que están basados en lattices. Estos lattices son grafos acíclicos dirigidos en los que se tiene una serie de fonemas (en general, sub-unidades de palabra) probables en cada etapa y en los que se pasa de un fonema de cada etapa a los fonemas de la etapa siguiente con una determinada probabilidad, como se puede observar en el ejemplo de la Figura 2.14, donde <s> y </s> son los silencios de principio y fin de tramo respectivamente y las letras mayúsculas representan fonemas.

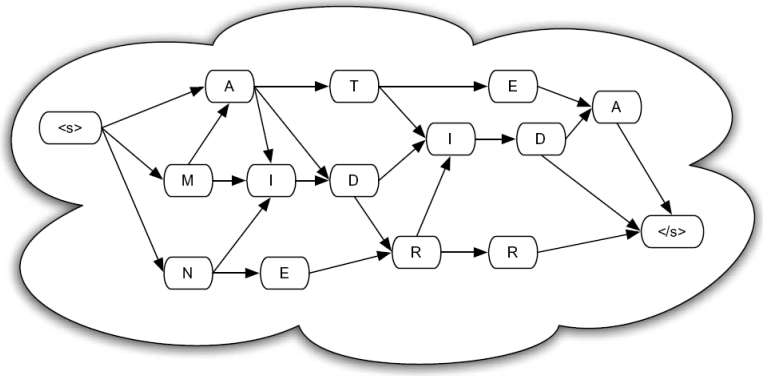

Figura 2.14: Ejemplo de un lattice de fonemas

Como se ha visto con anterioridad, estos sistemas están encuadrados dentro de los basados en reconocedores de sub-unidades de palabra, existiendo una primera fase off-line en la que el audio es indexado mediante un sistema de reconocimiento basado en fonemas generalmente y una segunda fase on-line en la que se realiza la búsqueda en base a esa indexación. Una tercera fase sería la toma de decisiones acerca de si las ocurrencias encontradas son aceptadas o no según su puntuación. Así lo define el National Institute of Standards and Technology (NIST) en su evaluación de STD de 2006 [22].

La gran cercanía en el tiempo de la aparición de una evaluación de STD por parte del NIST indica la juventud de esta técnica de reconocimiento de palabras clave. Debido a los resultados que se han ido obteniendo a lo largo de estos años gracias a esta técnica, se puede deducir la proyección futura de la misma.

Se dice que STD es una técnica de word spotting basada en lattices ya que la indexación obtenida en la primera fase de estos sistemas suele ser o bien una cadena de los fonemas más probables (1-best) o un grafo con los N fonemas más probables en cada etapa (N-best).

Una de las ventajas más destacables de los sistemas basados en lattices como STD es la rapidez en el reconocimiento de palabras clave frente a los sistemas basados en reconocedores de gran vocabulario (LVCSR). Esta rapidez se debe a que el reconocimiento es realizado una única vez en la primera fase del proceso. Para el idioma español, ese reconocimiento se suele realizar usando Modelos Ocultos de Markov (HMMs) basados en sub-unidades de palabra (generalmente fonemas).

El resultado de la evaluación de STD realizada por el NIST en 2006 muestra que los sistemas basados en gran vocabulario obtienen mejores resultados que los sistemas STD siempre y cuando los términos buscados pertenezcan al vocabulario. Por el contrario, esta evaluación también

muestra el mejor comportamiento de STD frente a los sistemas LVCSR en el caso de las palabras Out-Of-Vocabulary (OOV).

Precisamente esa capacidad que tienen los sistemas STD de lidiar con este tipo de palabras (Out-Of-Vocabulary) es otra de sus grandes cualidades. Esta gran ventaja se hace notar aún más si se piensa en audio proveniente de información multilingüe (de seguridad, por ejemplo) en la que se puede querer buscar términos en cualquier idioma, documentación técnica en la que suelen buscarse extranjerismos o simplemente en el hecho de que la mayor parte de las búsquedas en audio tienen que ver con nombres propios (París, Albert Einstein) o acrónimos (Renfe, OTAN). En definitiva, STD resulta un sistema muy útil a la hora de buscar términos en audio heterogéneo, algo que cada vez es más frecuente dada el reciente crecimiento de la popularidad de las búsquedas en contenidos de audio, principalmente ubicados en Internet.

Como en muchos otros aspectos de la tecnología, la mejor solución suele ser la combinación de sistemas, los sistemas híbridos. Así, los sistemas STD suelen ser el complemento perfecto a los sistemas LVCSR, usando estos últimos para las palabras dentro de vocabulario y los primeros para localizar las palabras fuera de vocabulario (OOV).

Como se ha comentado, lo más normal es usar reconocedores basados en fonemas en la primera fase de los sistemas STD. Sin embargo, existen varias versiones de STD en las que se han usado grafemas (letras) como base para el reconocimiento y se han comparado con el caso de los fonemas [34]. En el caso concreto del idioma español se obtienen resultados bastante buenos con el uso de grafemas ya que existe una correspondencia bastante regular de letras a sonidos.

2.6.2. Evaluación del rendimiento de un sistema STD

Para evaluar y mejorar el rendimiento de un sistema Spoken Term Detection (STD) necesitamos obtener una medida objetiva y cuantitativa del mismo. Con esta medida podremos comparar diferentes sistemas implementados en términos de rendimiento. Los errores que comete un sistema STD pueden proporcionarnos una magnitud que defina el rendimiento del sistema de forma representativa. Existen dos tipos de errores diferentes que puede cometer un sistema de esta naturaleza:

- Falsa Aceptación o Falsa Alarma (FA): Ocurre cuando se acepta erróneamente una secuencia de fonemas del lattice como coincidente con una de las palabras del diccionario de búsqueda.
- Falso rechazo ó pérdida (Miss): Ocurre cuando el sistema no cataloga una secuencia de fonemas del lattice como coincidente con una palabra del diccionario de búsqueda cuando en realidad lo era.

El comportamiento de estos tipos de error depende del umbral de decisión utilizado en la última fase del sistema. Con un umbral de decisión bajo, el sistema tiende a aceptar a muchas secuencias de fonemas como coincidentes con alguna palabra del diccionario de búsqueda dando lugar a pocas pérdidas y muchas falsas aceptaciones. Por otra parte, con un umbral de decisión alto, el sistema rechazará a la mayoría de secuencias de fonemas candidatas a ser coincidentes por lo que se producirán muy pocas falsas aceptaciones y muchas pérdidas.

La evaluación de los sistemas STD tiene dos características muy importantes [35]:

- La pérdida de un término es penalizada más duramente que una falsa alarma de ese término.

- Los resultados de la detección son promediados para todos los términos buscados independientemente de sus coincidencias, es decir, la evaluación considera la contribución de todos los términos por igual. Por lo tanto, en una búsqueda las palabras fuera de vocabulario (OOV) tienen el mismo valor que las que se encuentran en el vocabulario.

Existen tres formas muy comunes de representar el rendimiento de un sistema STD:

- Occurrence-Weighted Value (OCC): Se calcula asignando pesos a las coincidencias correctas y restándole los pesos de las falsas alarmas como se indica a continuación:

$$OCC = \frac{\sum_{terms} [VN_{correct}(t) - C_{FA}N_{FA}(t)]}{\sum_{terms} VN_{true}(t)}$$

  donde:

  - t es cada término del diccionario de búsqueda.
  - $N_{correct}(t)$ es el número de coincidencias encontradas del término t que son correctas.
  - $N_{FA}(t)$ es el número de falsas alarmas del término t.
  - $N_{true}(t)$ es el número real de veces que el término t aparece en el audio.
  - V es el peso que se le asigna a las coincidencias correctas.
  - C_{FA} es el peso que se le asigna a las falsas alarmas.

  Es un parámetro que tiene una tendencia inherente a tener más en cuenta los términos más probables (no sigue del todo las premisas de la evaluación de STD descritas anteriormente). Mayores valores de este parámetro indican mejores resultados.

- Actual Term-Weighted Value (ATWV): Se calcula promediando las probabilidades de pérdidas y falsas alarmas de la siguiente forma:

$$ATWV = 1 - \frac{\sum_{terms} [P_{Miss}(t) + \beta P_{FA}(t)]}{T}$$

  donde:

  - t es cada término del diccionario de búsqueda.
  - $P_{Miss}(t)$ es la probabilidad de pérdida del término t.
  - $P_{FA}(t)$ es la probabilidad de falsa alarma del término t.
  - T es el número total de términos que hay en el diccionario de búsqueda.
  - $\beta = \frac{C}{V}(P_{prior}^{-1}(t) - 1)$, donde:
    - C es el peso que se les asigna a las falsas alarmas.
    - V es el peso que se les asigna a las coincidencias correctas.
    - $\frac{C}{V}$ suele ser 0.1 ya que se le da más importancia (más peso) a las coincidencias correctas.
    - $P_{prior}(t)$ suele ser 10^{-4}
    - Con estos valores, β vale aproximadamente 1000.

  Este es el parámetro usado para la evaluación del sistema desarrollado en este proyecto. Mayores valores de este parámetro indican mejores resultados.

- Curva DET (Detection Error Trade-off): curva monótona y decreciente que representa el rendimiento de un sistema dibujando la probabilidad de Falsa Alarma, P(FA), como función de la probabilidad de pérdida, P(Miss), en una escala de desviación normal. Permite una comparación visual entre sistemas más clara y fácil de realizar. La distancia entre curvas expresa las diferencias entre rendimientos de manera más significativa. Cuanto más cerca está una curva del origen, más robusto será el sistema. Un ejemplo de una curva DET se puede observar en la Figura 2.15. Este tipo de curvas son usadas para observar el rendimiento del sistema desarrollado en este proyecto.

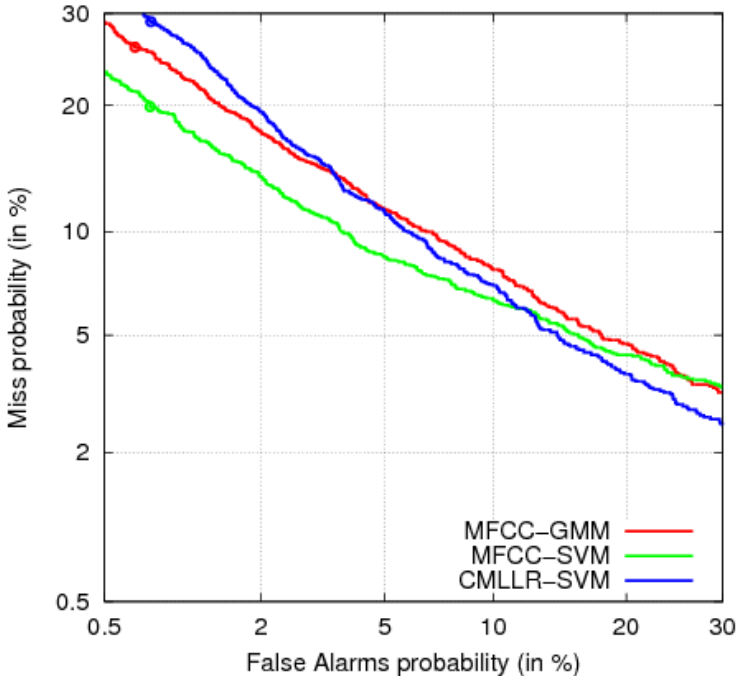

Figura 2.15: Ejemplos de curvas DET

3

Desarrollo del sistema

3.1. Introducción

Este capítulo tiene como objetivo realizar una descripción detallada del sistema desarrollado, así como analizar en profundidad cada uno de los elementos de los que se dispone para implementar el sistema.

Para alcanzar la meta de realizar búsquedas de palabras clave en los archivos que contienen el audio de los vídeos de información turística y evaluar dichas búsquedas, el sistema desarrollado tiene varios grandes bloques que se detallarán en profundidad más adelante:

- Reconocedor de voz: Este primer bloque sería el sistema desarrollado por José Antonio Morejón Saravia en su Proyecto Fin de Carrera 'Segmentación de audio y de locutores para recuperación de información multimedia y su aplicación a vídeos de información turística' [1]. Tiene como entrada el audio de la base de datos de vídeos de información turística y devuelve como resultado una serie de archivos de lattices que contienen los fonemas reconocidos.

- Detector de términos orales: Este es el sistema proporcionado por la Faculty of Information Technology de la Brno University of Technology (República Checa) y se llama LatticeSTD. Recibe como entrada básicamente los archivos de lattices devueltos por el sistema de reconocimiento de voz y un diccionario de términos a buscar (opcionalmente también recibe una matriz de confusión a aplicar). Devuelve un archivo con el resultado de la búsqueda donde se ve reflejado en qué archivos de lattices ha encontrado qué términos.

- Conversor de archivos necesarios para la evaluación: Este sistema consiste en una serie de scripts programados para convertir los archivos con los que se cuenta en la búsqueda (lista de archivos de lattices donde buscar, lista de palabras a buscar y la salida con los términos encontrados) en otros archivos con la misma información pero adaptados al formato que requiere la herramienta de evaluación que vamos a usar.

- Alineador de audio: Este bloque complementa al evaluador de búsquedas proporcionándole un archivo en formato RTTM con el alineamiento de todo el audio correspondiente a los archivos en los que se ha buscado.

- Evaluador de búsquedas de términos realizadas con Spoken Term Detection (STD): Esta herramienta es proporcionada por el National Institute of Standards and Technology (NIST). Recibe principalmente el resultado de la búsqueda realizada por el detector de términos orales y devuelve varios archivos, que más adelante se verán en detalle, de entre los cuales el principal es un archivo en el que se representan las estadísticas correspondientes a la evaluación de la búsqueda.

En la Figura 3.1 se observa como quedaría un diagrama de bloques del sistema completo. De todos estos bloques, este proyecto se centra en poner en marcha el detector de términos orales y evaluarlo con la herramienta del NIST, ayudado por el conversor de archivos y el alineador de audio.

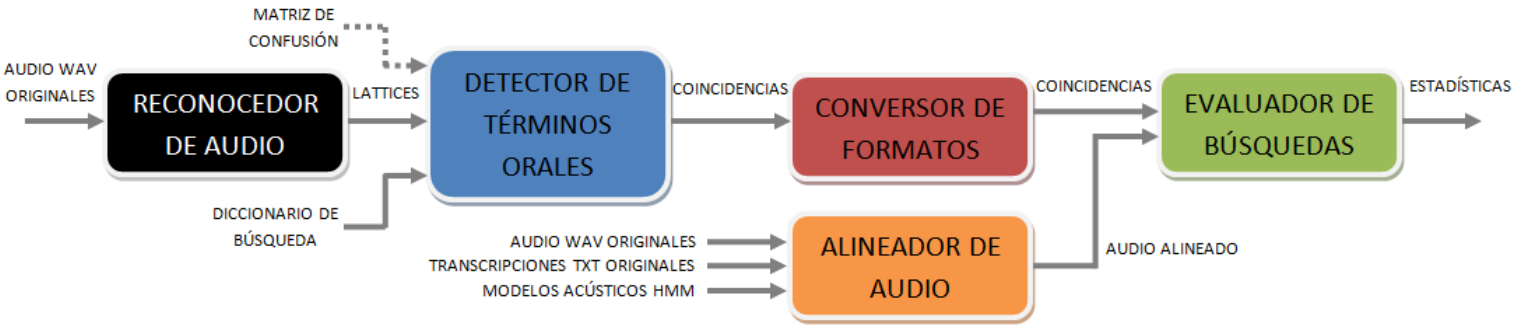

Figura 3.1: Diagrama de bloques del sistema completo

3.2. Medios disponibles

En primer lugar se va a hacer un análisis de los medios de los que se dispone previamente para realizar la parte del sistema completo en la que se centra este proyecto. Es importante destacar que en esta sección no se va a analizar la funcionalidad de los bloques de los que se compone el sistema completo, a pesar de que algunos de los medios disponibles coinciden con algunos de esos bloques. En esta sección se describirán los elementos con los que se contaba antes de desarrollar el sistema completo, dejando la explicación de la funcionalidad de los bloques para la siguiente sección.

3.2.1. Base de datos de audio

La base de datos de audio con la que se cuenta son los archivos del sonido correspondiente a los vídeos de información turística que se utilizan en el marco del proyecto MA2VICMR (Mejorando el Acceso, el Análisis y la Visibilidad de la Información y los Contenidos Multilingüe y Multimedia en Red) para la Comunidad de Madrid, que constituye la segunda edición del proyecto MAVIR.

El Consorcio MAVIR [36] es una red de investigación co-financiada por la Comunidad de Madrid y el Fondo Social Europeo bajo los programas de I+D en TIC MA2VICMR [37] (2010-2013) y MAVIR [38] (2006-2009) formada por un equipo multidisciplinar de científicos, técnicos, lingüistas y documentalistas para desarrollar un esfuerzo integrador en las áreas de investigación, formación y transferencia de tecnología.

El núcleo del consorcio está formado por siete grupos de investigación de universidades y centros de la Comunidad de Madrid que, desde una perspectiva pluridisciplinar, se complementan en varias dimensiones: mundo académico vs. mundo profesional, investigación vs. oferta de

Alcala_esp	Andalucia_eng	Andalucia_esp	Aranjuez_eng
Aranjuez_esp	Avila_esp	Baeza_eng	Baeza_esp
Bcngotico_eng	Bcngotico_esp	Bcnmodernista_eng	Bcnmodernista_esp
Caceres_esp	CaminoSantiago_esp	CastillaLeon_esp	Cid_esp
Cuenca_esp	Dalisurre_eng	Dalisurre_esp	Escorial_esp
Extremadura_eng	Extremadura_esp	Gaudi_eng	Gaudi_esp
GaudiGenio_eng	GaudiGenio_esp	Ibiza_esp	Laguna_esp
LaMancha_eng	LaMancha_esp	Madrid_eng	Madrid_esp
Montjuic_eng	Montjuic_esp	PueblosEdadMedia_esp	RomanicoAragon_esp
Salamanca_esp	SantiagoCompostela_esp	Segovia_eng	Segovia_esp
Toledo_eng	Toledo_esp	Ubeda_eng	Ubeda_esp
Valencia_eng	Valencia_esp		

Tabla 3.1: Archivos de audio originales (WAV) de la base de datos de MA2VICMR

servicios, generación de recursos vs. aplicaciones. Los centros a los cuales pertenecen esos grupos son:

- Centro Superior de Investigaciones Científicas (CSIC) [39]
- Universdad Autónoma de Madrid (UAM) [40]
- Universidad Carlos III de Madrid (UC3M) [41]
- Universidad Europea de Madrid (UEM) [42]
- Universidad Nacional de Educación a Distancia (UNED) [43]
- Universidad Politécnica de Madrid (UPM) [44]
- Universidad Rey Juan Carlos (URJC) [45]

La base de datos se compone de 46 archivos WAV correspondientes al audio de los vídeos de información turística en los que existe tanto voz como música. Dichos archivos cuentan con un solo canal y están muestreados a 16 kHz con 16 bits/muestra. La primera parte del nombre de estos archivos indica la ciudad, región o arte sobre el que tratan. Existen 30 archivos en idioma español y 16 en idioma inglés (distinguidos por el sufijo 'esp' o 'eng' en el nombre del archivo, respectivamente). En este proyecto únicamente se van a utilizar los archivos en idioma español ya que son de los únicos de los cuales se dispone de sus correspondientes lattices de fonemas reconocidos así como de las transcripciones originales.

En la Tabla 3.1 se proporciona una relación de los 46 archivos de audio que conforman la base de datos. En el proyecto de José Antonio Morejón Saravia [1], estos audios son el punto de partida para el reconocimiento que da lugar a los lattices. En este proyecto, estos archivos de audio han sido usados junto con las transcripciones para obtener el alineamiento que requiere la herramienta de evaluación.

3.2.2. Transcripciones originales

Se cuenta también con las transcripciones originales de los audios en idioma español anteriormente descritos. Estas transcripciones consisten en archivos TXT con un formato específico en donde se concreta por tramos de qué instante de tiempo a qué otro instante de tiempo va cada

Alcala_esp	Andalucia_esp	Aranjuez_esp	Avila_esp
Baeza_esp	Bcngotico_esp	Bcnmodernista_esp	Caceres_esp
CaminoSantiago_esp	CastillaLeon_esp	Cid_esp	Cuenca_esp
Dalisurre_esp	Escorial_esp	Extremadura_esp	Gaudi_esp
GaudiGenio_esp	Ibiza_esp	Laguna_esp	LaMancha_esp
Madrid_esp	Montjuic_esp	PueblosEdadMedia_esp	RomanicoAragon_esp
Salamanca_esp	SantiagoCompostela_esp	Segovia_esp	Toledo_esp
Ubeda_esp	Valencia_esp		

Tabla 3.2: Transcripciones originales (TXT) del audio de la base de datos de MA2VICMR

frase y cada fragmento de música del audio correpondiente. Existen 30 transcripciones nombradas con el mismo nombre que sus audios correspondientes (el tema del que tratan seguido de 'esp') que han sido realizadas a mano. La relación de todos las transcripciones de las que se dispone se puede ver en la Tabla 3.2 y un ejemplo de ellas se muestra en la Figura 3.2.

El formato de estos archivos es parecido a XML y es el siguiente:

- Información sobre los grupos que componen el consorcio MAVIR.
- Cabecera: en ella se especifican el tema sobre el que trata la información contenida en el correspondiente audio, las palabras clave, duración, número de palabras, información sobre el locutor e información sobre la música de fondo y la calidad del audio.
- Cuerpo: se compone de dos grandes bloques: prosodia y ortografía. Ambos bloques representan prácticamente lo mismo y con el mismo formato. En ellos se puede ver el texto de los tramos de voz (denotados por el identificador 'UNIT') detallándose el instante de inicio y de fin en segundos (ambos con tres decimales) y el locutor. También se ven los tramos de música con sus instantes de inicio y fin. La diferencia entre el bloque de prosodia y el de ortografía es que en el de prosodia se especifican las pausas y silencios que hay entre las frases en los tramos de voz, denotados por barras o dobles barras.
- Cola: en ella se reserva un espacio para introducir metainformación sobre el audio correspondiente.

Para este proyecto en concreto, estas transcripciones son las que han dado pie junto con el audio a obtener el alineamiento del mismo, necesario para la evaluación de los resultados de las búsquedas.

3.2.3. Lattices de fonemas reconocidos

Estos lattices son la herencia obtenida del Proyecto Fin de Carrera 'Segmentación de audio y de locutores para recuperación de información multimedia y su aplicación a vídeos de información turística' realizado por José Antonio Morejón Saravia [1] en el que entre otras cosas se realizó la segmentación de toda la base de datos de audio anteriormente descrita. También se realizó la separación entre segmentos de voz y de música. Del reconocimiento de esos segmentos de voz nacen estos lattices en los que se refleja el grafo acíclico dirigido de fonemas y sus probabilidades. Los lattices de fonemas reconocidos son el punto de partida de este proyecto.

Como bien se ha comentado anteriormente, en este proyecto sólo se trabaja con la parte de la base de datos correspondiente al idioma español, debido a que sólo se cuenta con las

Figura 3.2: Ejemplo de la estructura de una transcripción original

transcripciones de los archivos de audio en español. Por cada uno de los 30 archivos de audio con los que se cuenta se tienen múltiples archivos de lattices siguiendo siempre el mismo patrón: existe un lattice cada cuatro segundos de audio con un solape entre ellos de dos segundos. En la Figura 3.3 se observa gráficamente cómo están organizados los lattices dentro del audio.

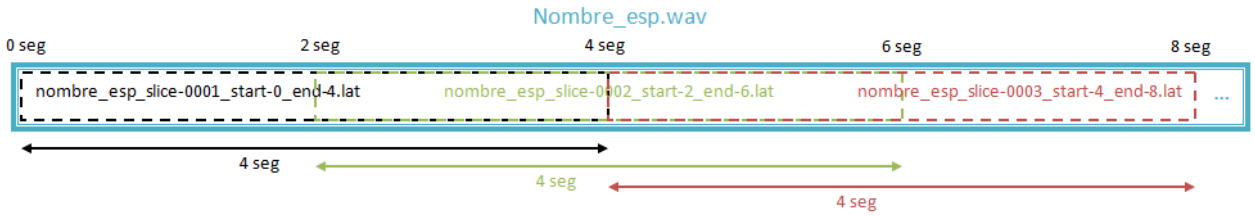

Figura 3.3: Estructura de lattices dentro de un archivo de audio

Los archivos de lattices tienen extensión LAT (aunque se verá más adelante que después es necesario comprimirlos) y su nomenclatura es muy clara. Cada uno de ellos lleva por nombre el tema del que trata seguido de 'esp' y seguido del número de porción de audio o slice al que corresponde y los segundos que abarca, de la siguiente forma:

nombre_ esp_ slice-XXXX_ start-XX_ end-XX.lat

La estructura interna de los lattices sigue el siguiente patrón:

- Cabecera: contiene información sobre el audio y su reconocimiento. En la Figura 3.4 se puede observar un ejemplo de cabecera.

```
VERSION=1.0
UTTERANCE=../../../Audios/CortadosHTK/Alcala_esp/Alcala_esp_slice-0001_start-0_end-4.wav
lmname=../RecFon/wdNetBigramaFinalEntrenamientoFonemasAlbayzinSilIniSilFinSp
lmscale=3.00   wdpenalty=-15.00
acscale=1.00
vocab=../RecFon/dictAlbayzinSilIniSilFinSp.txt
N=17380 L=73192
I=0    t=0.00  W=!NULL
I=1    t=0.02  W=R1                    v=1
I=2    t=0.02  W=R1                    v=1
```

Figura 3.4: Ejemplo de cabecera de un archivo de lattices

- Fonemas: en esta parte del lattice se representa un nodo del grafo por cada línea. Como se ve en la Figura 3.5, para cada uno se pueden observar los siguientes campos:

- Un identificador que indica el orden de los nodos dentro del archivo LAT (denotado por 'I=').
- El instante (dentro de los cuatro segundos de audio a los que corresponde ese lattice) para el que ese fonema es probable (denotado por 't=').
- El fonema que indica ese nodo del grafo (denotado por 'W=').
- La versión, que en este caso siempre es la 1 (denotado por 'v=').

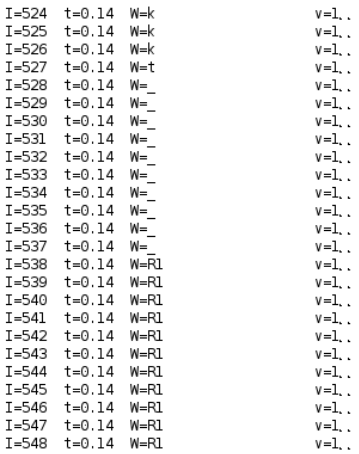

Figura 3.5: Ejemplo del tramo de fonemas de un archivo de lattices

- Probabilidades: en esta parte del lattice se encuentran las probabilidades de transición entre los distintos fonemas que componen el grafo acíclico dirigido. Se puede ver un ejemplo de este tramo en la Figura 3.6. El valor denotado por 'a=' es la probabilidad dada por el modelo acústico utilizado y el valor denotado por 'l=' es la probabilidad dada por el modelo de lenguaje.

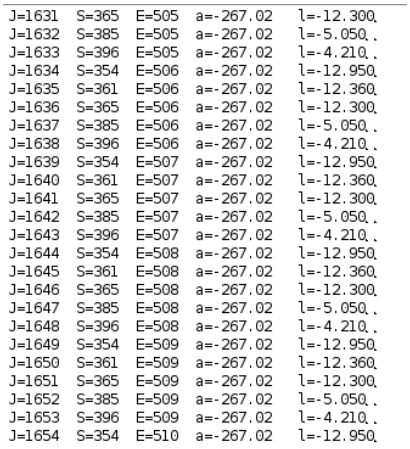

Figura 3.6: Ejemplo del tramo de probabilidades de un archivo de lattices

Los lattices con los que se cuenta en un primer momento no son del todo adecuados para su uso en este proyecto ya que entre sus fonemas reconocidos existe una muy amplia variedad. Esto supone un inconveniente ya que hay muchos casos en los que tratándose del mismo sonido, éste está representado por distintos fonemas en los lattices. Por ejemplo, el sonido 'a' se puede encontrar representado por los fonemas 'a', 'A', 'an' ó 'An' como se ve en la Figura 3.7. Esto

se debe a que cada una de las variedades de la 'a' representa cada uno de los posibles sonidos, pronunciaciones o alófonos del fonema /a/.

Donde más importa este inconveniente es a la hora de realizar un diccionario de búsqueda donde, como ya se verá, es necesario escribir la transcripción fonética de los términos a buscar. Si tenemos varios posibles fonemas correspondientes al sonido 'a', por ejemplo, habría que escribir cada palabra que contenga un sonido 'a' con todas sus posibles opciones, cosa que no es nada cómoda.

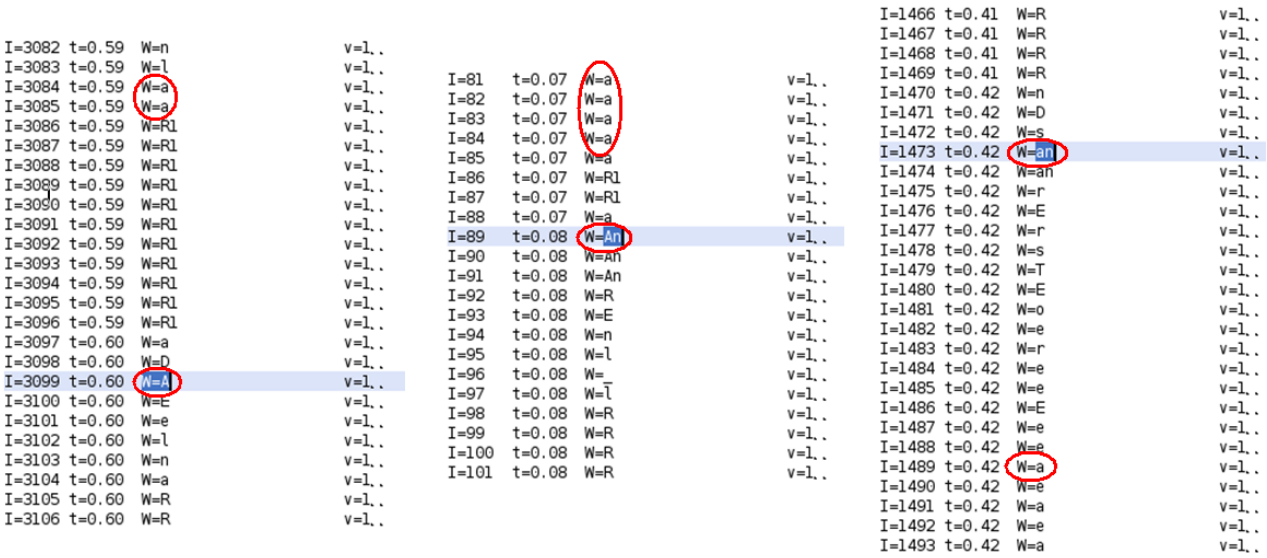

Figura 3.7: Ejemplos de aparición de un mismo fonema representado por distintos símbolos

Por lo tanto, una primera acción previa a comenzar a trabajar con los lattices ha sido la simplificación de los mismos de manera que se redujera a 25 el número de fonemas distintos utilizados en ellos, que se corresponden aproximadamente con los fonemas más comunes del idioma español. Para ello, se ha hecho uso de la aplicación SED con la que mediante un archivo de conversión (cuya función se ve reflejada en la Tabla 3.4) se simplifican los fonemas.

De esta forma se obtiene un nuevo archivo de lattice por cada archivo de lattice del reconocimiento original. La diferencia entre ambos se observa en la Figura 3.8 y Figura 3.9.

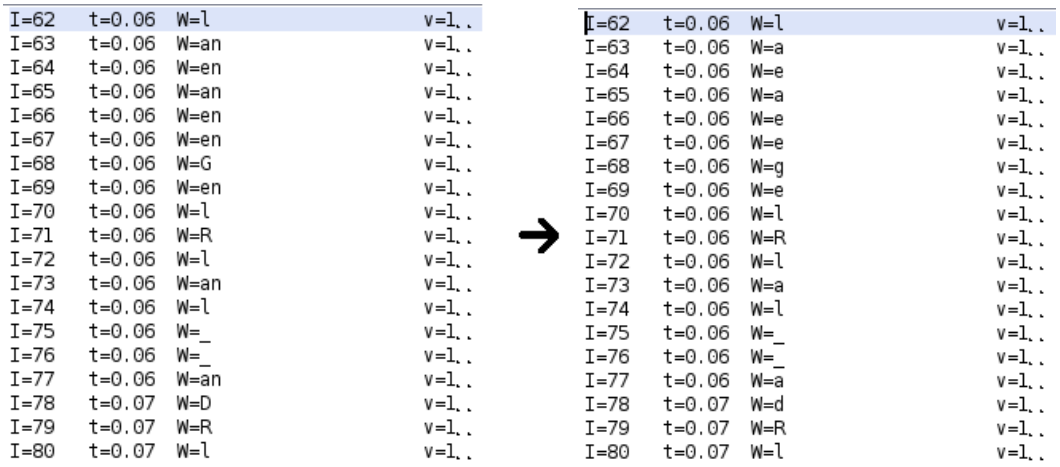

Figura 3.8: Lattice antes de ser simplificado

Figura 3.9: Lattice tras ser simplificado

3.2.4. Herramienta LatticeSTD

Esta herramienta es una aplicación desarrollada en la Faculty of Information Technology de la Brno University of Technology (República Checa). Está desarrollada en C++, lenguaje de programación orientada a objetos, paradigma de programación en el que las unidades son las clases. De estas clases se pueden crear objetos y cada una cuenta con sus propios métodos o funciones. La aplicación se encarga, en general, de encontrar términos en archivos de lattices de fonemas. Más en particular, se compone de una serie de archivos fuente con extensión CPP de entre los cuales los más importantes son:

- Latt2Multigram.cpp (main): Desde él se ejecuta la función principal del programa en la que se realizan diversas acciones:
  - Almacena los parámetros elegidos por el usuario para la búsqueda.
  - Configura todo lo relativo a dichos parámetros (matriz de confusión, permisividad de inserciones, borrado y sustituciones, etc).
  - Mediante un bucle, recorre todos los archivos de la lista de archivos en los que debe buscar términos, buscando coincidencias en ellos con los términos del diccionario.
  - Libera toda la memoria usada.
- DetectionFilter.cpp: En este archivo se definen las clases relacionadas con la detección de coincidencias así como con el cálculo del score de las mismas.
- Latt2MGram.cpp: En este archivo se encuentra la clase C_FindMultiGramInLattice dentro de la cual se definen los métodos necesarios para realizar la búsqueda de coincidencias en los lattices. Los métodos más importantes son:
  - Search_ExactMatch: Busca coincidencias de una palabra del diccionario de búsqueda dentro del lattice.
  - Search_ExactMatchRecursive: Usa la función anterior recursivamente para buscar términos y si encuentra más de una coincidencia calcula un score que dará una idea de cuales de ellas serán mejores o peores.
- argvparser.cpp: En este archivo se encuentra la clase ArgvParser cuyos métodos realizan la recogida y almacenamiento de los argumentos que define el usuario al configurar los parámetros del programa.
- ConfMatrix.cpp: En este archivo se encuentra la clase C_ConfusionMatrix cuyos métodos se encargan de todo lo relacionado con la matriz de confusión, en caso de existir.
- PronDictTree.cpp y PronDictTreeTokenPass.cpp: En estos archivos se encuentran las clases C_PronunciationDictionaryTree y C_PronunciationDictionaryTreeTokenPass que ayudan a la búsqueda de términos dentro de los lattices, manejando los nodos de éstos.

En el Anexo D se muestra una estructura de archivos fuente, clases y métodos de esta aplicación.

3.2.5. Herramienta evaluación NIST

Esta aplicación se denomina STDEval-0.7 y se puede descargar directamente desde la página que el NIST (National Institute of Standards and Technology) tiene dedicada a Spoken Term Detection [46]. Su misión es evaluar las búsquedas que se realicen con un sistema de STD, en este caso, con el sistema LatticeSTD anteriormente descrito.

El sistema se compone de varios scripts en lenguaje PERL de entre los cuales el más importante es STDEval.pl donde se ejecuta el hilo principal de la aplicación y desde donde se llama a los demás scripts. Este proyecto no se ha detenido en analizar en profundidad las entrañas de esta aplicación ya que proviene de una fuente fiable como es el NIST y lo que más nos interesa es su funcionalidad que se explica en la siguiente sección.

3.2.6. Modelos acústicos de idioma español

Estos modelos son los Modelos Ocultos de Markov o Hidden Markov Models (HMMs) que se usan en la etapa de alineamiento del audio de cara a la evaluación de las búsquedas. Han sido proporcionados por el tutor de este proyecto Doroteo Torre Toledano y provienen de los sistemas de reconocimiento de voz utilizados por el Área de Tratamiento de Voz y Señales (ATVS) de la Universidad Autónoma de Madrid (UAM), grupo donde se ha realizado este proyecto.

Más en concreto en los archivos proporcionados existen varios modelos distintos: para inglés y español, para 8 y 16 kHz (estos últimos sólo para español) y para distintos números de Gaussianas (1 a 4).

3.3. Sistema completo desarrollado

En esta sección se van a desglosar los distintos bloques del sistema completo, detallando las entradas y salidas de cada bloque así como los subbloques en los que se divide cada uno, si procede. La sección se va a centrar en los bloques que se han desarrollado y/o puesto en marcha en este proyecto. En la Figura 3.10 se puede ver un diagrama en el que se muestra la interconexión de los bloques en los que se centra el proyecto.

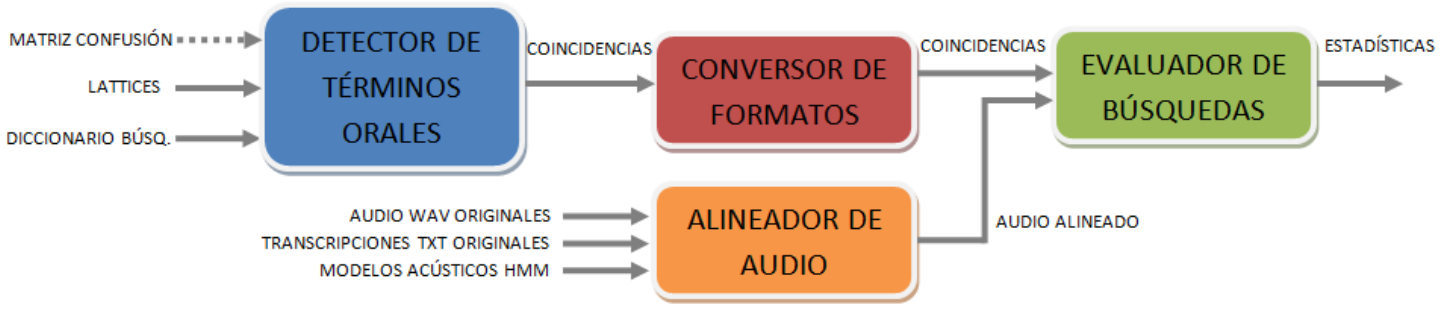

Figura 3.10: Diagrama de bloques de los sistemas desarrollados en este proyecto

3.3.1. Detector de términos orales

Este sistema, del que ya se han analizado los detalles internos en la sección anterior, se encarga de encontrar términos orales en los lattices de fonemas del audio reconocido correspondiente a la base de datos también descritas en la anterior sección.

3.3.1.1. Entradas y salidas del sistema

Las entradas de este sistema son las siguientes:

- Lista de lattices: Consiste en un archivo de texto en el que en cada línea se cita un archivo de lattices de fonemas en el que se desea buscar términos. La lista debe tener extensión LIST y en él se deben especificar las rutas completas a los archivos de lattices deseados. Dichos archivos de lattices deben tener extensión LAT.GZ, es decir, deben estar comprimidos. En la Figura 3.11 se muestra un extracto de uno de estos archivos de lista.

```
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0001_start-0_end-4.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0002_start-2_end-6.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0003_start-4_end-8.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0004_start-6_end-10.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0005_start-8_end-12.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0006_start-10_end-14.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0007_start-12_end-16.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0008_start-14_end-18.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0009_start-16_end-20.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0010_start-18_end-22.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0011_start-20_end-24.lat.gz
/home/pablo/Lattices_sin_adapt/Alcala_esp_slice-0012_start-22_end-26.lat.gz
```

Figura 3.11: Lista de lattices (LIST)

- Diccionario de búsqueda: Consiste en un archivo de texto en el que en cada línea se cita un término a buscar y su correspondiente transcripción fonética. El diccionario debe tener extensión DICT y se pueden escribir en él tantos términos como se quiera. En la Figura 3.12 se puede ver un ejemplo de diccionario de búsqueda. Para configurar las transcripciones fonéticas de los términos se hace uso del juego de fonemas del idioma español que se puede observar en la Tabla 3.3.

```
universidad.    u n i b e r s i d a d
renacentistas.  R e n a T e n t i s t a s
arquitectura.   a r k i t e k t u r a
restaurantes.   R e s t a u r a n t e s
ayuntamiento.   a y u n t a m i e n t o
barcelona.      b a r T e l o n a
modernismo.     m o d e r n i s m o
plateresca.     p l a t e r e s k a
compostela.     k o m p o s t e l a
hospitales.     o s p i t a l e s
```

Figura 3.12: Diccionario de búsqueda (DICT)

Fonema	/a/	/b/	/T/	/C/	/d/	/e/	/f/	/g/	/i/	/j/	/k/	/l/	/L/
Letras que representa	a	b, v	c, z	ch	d	e	f	g	i	j	c, k, q	l	ll

Fonema	/m/	/n/	/N/	/o/	/p/	/r/	/R/	/s/	/t/	/u/	/x/	/y/
Letras que representa	m	n	ñ	o	p	r	rr	s	t	u, w	x	y

Tabla 3.3: Juego de fonemas usado para la transcripción de los términos de búsqueda

- Matriz de confusión: Consiste en un archivo de texto en el que se especifica la matriz de confusión que se desea aplicar a las búsquedas. En una matriz de confusión se dicen las

probabilidades de inserción, borrado y sustitución de cada fonema con todos los demás (él mismo incluido). El formato de las matrices de confusión que utiliza LatticeSTD (del cual muestra un ejemplo la Figura3.13) es el siguiente:

1. Número de fonemas existentes en la matriz.
2. Lista completa de fonemas.
3. Matriz:
   - En cada fila está la probabilidad de sustitución de cada fonema por el resto salvo en la última fila que se muestra la probabilidad de inserción de cada fonema.
   - En cada columna está la probabilidad de que el resto de fonemas sean sustituidos por cada uno de los fonemas excepto la última columna que son las probabilidades de borrado.

La matriz de confusión correspondiente al idioma español utilizada en este proyecto es una matriz extraída del Proyecto Fin de Carrera de Doroteo Torre Toledano (profesor doctor de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid y tutor de este proyecto) 'Estimación a priori del grado de confusión entre palabras para reconocedores fonéticos de vocabulario flexible: métodos y aplicaciones' [47]. En el Anexo B se muestra la matriz de confusión completa utilizada.

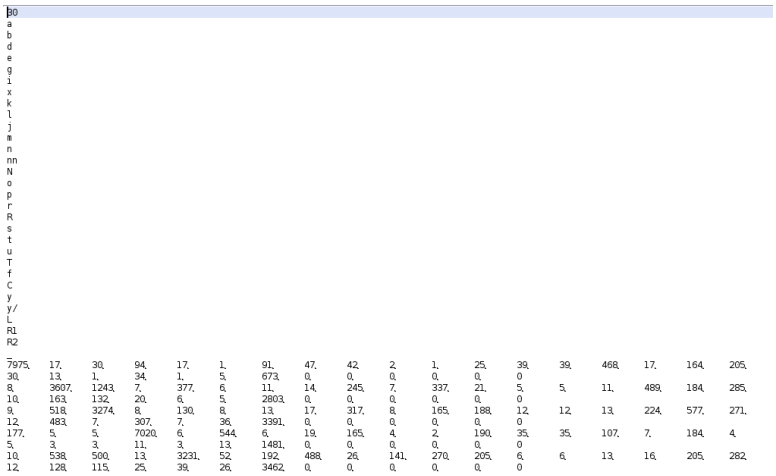

Figura 3.13: Extracto de la matriz de confusión

La salida de este sistema es:

- Coincidencias encontradas: Consiste en un archivo de texto en el que se enumeran todos los archivos de lattices en los que se ordenó buscar y debajo de cada uno de ellos se citan las palabras encontradas en él de entre las que forman el diccionario de búsqueda. Para cada una de las palabras encontradas en cada lattice se detalla el instante de inicio, el instante de fin y el score calculado para dicha coincidencia. Los instantes de inicio y fin están expresados en formato HTK, es decir, en unidades de cientos de nanosegundos, de otra forma:

$$instante(seg) = instante(formatoHTK) * 10^{-7}$$

y los scores son valores negativos de forma que cuanto más negativo es el número, peor puntuación tiene dicha coincidencia. Los valores de score no están normalizados pero esto

no es ningún impedimento ya que la herramienta de evaluación que tratará con estos scores está preparada para que éstos estén en cualquier formato. Este archivo tiene extensión MLF y un extracto de su estructura se representa en la Figura 3.14.

```
"*/Alcala_esp_slice-0017_start-32_end-36.rec"
9700000 12100000 úbeda -170.6110289404
.
"*/Alcala_esp_slice-0018_start-34_end-38.rec"
1700000 3400000 león -211.1247996762
20300000 22400000 león -154.9066310990
32500000 36400000 madrid -225.7223594206
35000000 38100000 úbeda -283.6723894842
.
"*/Alcala_esp_slice-0019_start-36_end-40.rec"
300000 2400000 león -344.6471881444
15700000 17200000 león -141.1071025930
35500000 37300000 ávila -149.8936305049
31800000 34400000 úbeda -111.7488886134
.
"*/Alcala_esp_slice-0020_start-38_end-42.rec"
5900000 7600000 león -54.9373806721
17600000 19700000 ávila -105.1985420985
11800000 14400000 úbeda -111.7492216130
.
```

Figura 3.14: Archivo de coincidencias (MLF)

3.3.1.2. Estructura interna de bloques

En la Figura 3.15 se observa la estructura de este sistema. Como se ve, existen dos pequeños subbloques previos al bloque principal LatticeSTD:

- Un primer subbloque consistiría en una llamada a la herramienta SED de cara a la simplificación de fonemas que se mencionó en el apartado 3.2.3 de la anterior sección, donde se hablaba de los lattices. La conversión de fonemas realizada por el archivo llamado por la herramienta SED se muestra en la Tabla 3.4 (sólo se muestras los fonemas que han sufrido conversión).

Fon. originales	a,A,an,An	b,B	T/	d,D	e,E,en,En	g,G
Fon. simplificado	a	b	C	d	e	g
i,I,in,In,j	X	N,Nn,nn	o,O,on,On	u,U,un,Un,w	gs	y,y/,J,J/
i	j	N	o	u	x	y

Tabla 3.4: Conversión de fonemas para la simplificación de los lattices

- Y un segundo subbloque sería un script cuya funcionalidad es comprimir los archivos de lattices en formato GZ para que sea el adecuado para la herramienta de búsqueda.

Figura 3.15: Estructura interna del detector de términos orales

3.3.1.3. Llamada al bloque

La llamada a este bloque tiene el siguiente formato:

./Latt2MultiGram
–list-file lista.list
–dictionary-file diccionario.dict
–filter-method BestTimeBestScore
–score-method LogLikeliBaumWelsh
–trace 0
–print-params
–print-conftable
–primary-tip -1
–primary-lmsf 1
–primary-acsf 1
–primary-kwsf 0.0001
–MLF-file salida.mlf
–confusionmatrix-file confusion_matrix_sp.txt
–allow-substitutions 0.1
–allow-insertions 0.1
–allow-deletions 0.1

En ella se llama al archivo fuente C++ desde el que se realiza la función principal del programa (Latt2MultiGram) y se especifican las distintas opciones:

- list-file: Lista de entrada con los nombres de los archivos de lattice que van a ser procesados.
- dictionary-file: Diccionario de términos a buscar.
- filter-method: Método elegido a la hora de filtrar las coincidencias encontradas en los lattices. Existen varias posibilidades para este campo:
  - NoFiltering
  - BestTimeBestScore
  - GroupTimeBestScore
  - BestTimeLogAddScore
  - BestTimeLogAddScoreCenter
  - BestTimeContinuousLogAddScore
- score-method: Método elegido para el cálculo del score de las coincidencias. Este campo puede tomar distintos valores:
  - LogLikeli
  - LogLikeliVitRatio
  - LogLikeliBaumWelsh
- trace: Parámetro de trazado.
- print-params: Indica que se desea imprimir los parámetros configurados por pantalla al ejecutar el programa.
- print-conftable: Indica que se desea imprimir la matriz de probabilidades de confusión.

- primary-tip: Primary token insertion penalty.
- primary-lmsf: Primary language model scaling factor.
- primary-acsf: Primary acoustic scaling factor.
- primary-kwsf: Primary keyword scaling factor.
- MLF-file: Archivo MLF en el que introducirá las coincidencias encontradas.
- confusionmatrix-file: Matriz de confusión de entrada.
- allow-substitutions: Permitir sustituciones según lo que indica la matriz de confusión.
- allow-insertions: Permitir inserciones según lo que indica la matriz de confusión.
- allow-deletions: Permitir borrados según lo que indica la matriz de confusión.

De todos estos parámetros, hay dos que son requisito indispensable en la llamada a LatticeSTD: list-file y dictionary-file.

3.3.2. Conversor de archivos necesarios para la evaluación

Este bloque tiene la funcionalidad de convertir el archivo de salida y algunos de los archivos de entrada del buscador de términos orales LatticeSTD (bloque inmediatamente anterior del que se habla en el apartado 3.3.1.) en otros archivos con un formato adecuado que requiere la herramienta de evaluación STDEval-0.7 (bloque inmediatamente posterior que se tratará en uno de los siguientes apartados).

3.3.2.1. Entradas y salidas del sistema

Las entradas de este sistema son:

- Archivo con extensión NAME con la lista de los archivos de audio en los que se ha buscado. Es importante poner especial énfasis en que este archivo no es la lista de lattices que se usaba como entrada en el buscador de términos y que contenía todas las rutas de los archivos de lattices (cada uno de ellos de cuatro segundos de duración). Este archivo es una lista con la ruta de los audios completos (en formato WAV) que se han usado en la búsqueda. Un extracto de este tipo de archivo se observa en la Figura 3.16.

```
/home/pablo/CorpusMAVIR/Audios_Originales/Alcala_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Andalucia_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Aranjuez_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Avila_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Baeza_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Bcngotico_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Bcnmodernista_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/Caceres_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/CaminoSantiago_esp.wav
/home/pablo/CorpusMAVIR/Audios_Originales/CastillaLeon_esp.wav
```

Figura 3.16: Extracto de lista de archivos de audio (archivo NAME)

- Archivo con extensión TIME que contiene una lista con la duración en segundos de los archivos de audio usados en la búsqueda. Tiene una trazabilidad clara con el archivo anteriormente mencionado (NAME) ya que cada línea del archivo TIME es la duración del archivo de audio (WAV) listado en esa misma línea del archivo NAME. Un extracto de este tipo de archivo se observa en la Figura 3.17.

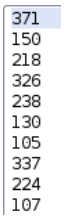

Figura 3.17: Extracto de lista de tiempos de los audios (archivo TIME)

- Diccionario de búsqueda (con extensión DICT) usado en la búsqueda con LatticeSTD. En él se especifican los términos a buscar y su transcripción fonética. Se puede observar un ejemplo de este tipo de archivos en la Figura 3.12.
- Resultados de la búsqueda. Es el archivo con formato MLF devuelto por LatticeSTD tras la búsqueda. Se puede observar un ejemplo de este tipo de archivos en la Figura 3.14.

Las salidas de este sistema son:

- Archivo con extensión ECF.XML en el que se ven reflejados, con una estructura tipo XML, los archivos de audio que se usan en la búsqueda. En la Figura 3.18 se presenta un tramo de un archivo ECF. El formato de este archivo es el siguiente:

  1. Cabecera (denotada por la partícula <ecf>): en ella se indica la suma de la duración de todos los audios presentes en la búsqueda y la versión.
  2. Cuerpo: cada línea (denotada por la partícula <excerpt>) contiene la ruta (proviene de la información contenida en el archivo NAME de la entrada), el canal, el tiempo de inicio, la duración (contenido en el archivo TIME de la entrada), el idioma y el tipo de fuente de cada archivo de audio que interviene en la búsqueda.

Figura 3.18: Extracto de archivo ECF

- Archivo con extensión TLIST.XML en el que se ve reflejada la lista de términos que se deseaba encontrar en la búsqueda con una estructura tipo XML. En la Figura 3.19 se observa un ejemplo de este tipo de archivo. El formato de los archivos TLIST es el siguiente:

  1. Cabecera (denotada por la partícula <termlist>): en ella se indica el archivo ecf con el que se corresponde este diccionario, la versión y el idioma.
  2. Cuerpo: cada línea (denotada por la partícula <term>) se refiere a cada uno de los términos de búsqueda contenidos en el archivo DICT de entrada. Se especifica un identificador de término y el propio término.

```
<termlist ecf_filename="30_largas.ecf.xml" version="001" language="spanish">
<term termid="TEST-01"><termtext>universidad</termtext></term>
<term termid="TEST-02"><termtext>renacentistas</termtext></term>
<term termid="TEST-03"><termtext>arquitectura</termtext></term>
<term termid="TEST-04"><termtext>restaurantes</termtext></term>
<term termid="TEST-05"><termtext>ayuntamiento</termtext></term>
<term termid="TEST-06"><termtext>barcelona</termtext></term>
<term termid="TEST-07"><termtext>modernismo</termtext></term>
<term termid="TEST-08"><termtext>plateresca</termtext></term>
<term termid="TEST-09"><termtext>compostela</termtext></term>
<term termid="TEST-10"><termtext>hospitales</termtext></term>
<term termid="TEST-11"><termtext>guadalajara</termtext></term>
<term termid="TEST-12"><termtext>cristianos</termtext></term>
<term termid="TEST-13"><termtext>delirante</termtext></term>
<term termid="TEST-14"><termtext>guadarrama</termtext></term>
<term termid="TEST-15"><termtext>extremadura</termtext></term>
<term termid="TEST-16"><termtext>cantábrico</termtext></term>
<term termid="TEST-17"><termtext>mudéjares</termtext></term>
<term termid="TEST-18"><termtext>mediterráneo</termtext></term>
<term termid="TEST-19"><termtext>tenerife</termtext></term>
<term termid="TEST-20"><termtext>cervantino</termtext></term>
<term termid="TEST-21"><termtext>pinacotecas</termtext></term>
<term termid="TEST-22"><termtext>catalunya</termtext></term>
<term termid="TEST-23"><termtext>albarracín</termtext></term>
<term termid="TEST-24"><termtext>capiteles</termtext></term>
<term termid="TEST-25"><termtext>salamanca</termtext></term>
<term termid="TEST-26"><termtext>jerusalén</termtext></term>
<term termid="TEST-27"><termtext>acueducto</termtext></term>
<term termid="TEST-28"><termtext>isabelino</termtext></term>
<term termid="TEST-29"><termtext>agricultura</termtext></term>
<term termid="TEST-30"><termtext>velázquez</termtext></term>
</termlist>
```

Figura 3.19: Ejemplo de archivo TLIST

- Archivo con extensión STDLIST.XML en el que se ven reflejados los resultados de la búsqueda con una estructura tipo XML. En la Figura 3.20 se muestra un extracto de este tipo de archivo. El formato de un archivo STDLIST es:

  1. Cabecera (denotada por la partícula <stdlist>): en ella se indica el archivo TLIST correspondiente a la búsqueda que produjo esa salida, el idioma y distintos identificadores.
  2. Cuerpo: en lugar de estar organizado por archivos y dentro de cada archivo las palabras encontradas en él de entre las contenidas en el diccionario de búsqueda, en esta ocasión está organizado por términos y dentro de ellos se encuentran los archivos en los que dicha palabra ha tenido coincidencias. Toda esa información se extrae del archivo MLF de entrada a este bloque y sigue la siguiente estructura:
     a) Término (denotado por la partícula <detected_termlist>): se indica el identificador de término siguiendo con lo establecido en el TLIST y otros dos campos sin relevancia.
     b) Coincidencia (denotada por la partícula <term>): se detalla nombre del archivo en el que se produjo la coincidencia, canal, instante de inicio, duración, score (de valor negativo, cuanto más negativo peor es la coincidencia) y decisión que es un campo en el que se dice si esa ocurrencia del término es verdadera o no (toma los valores YES o NO), más adelante -en el capítulo 4- se verá cómo este campo puede servir para mejorar los resultados.

  Todos los términos cuentan con un bloque <detected_termlist> aunque dentro de ellos a veces no haya bloques <term> de coincidencias.

3.3.2.2. Estructura interna de bloques

Este bloque cuenta con varios subsistemas que forman una estructura que se puede observar en la Figura 3.21. Cada subsistema se corresponde con la conversión de un tipo de archivos de los necesarios y con uno de los archivos fuente en lenguaje C que componen el sistema.

```
<stdlist termlist_filename="30_largas.tlist.xml" indexing_time="1.00" language="spanish" index_size="1" system_id="MAVIR">
<detected_termlist termid="TEST-01" term_search_time="24.3" oov_term_count="0">
<term file="Alcala_esp" channel="1" tbeg="86.690" dur="0.750" score="-57.888176" decision="YES"/>
<term file="Alcala_esp" channel="1" tbeg="163.560" dur="0.650" score="-108.467311" decision="YES"/>
<term file="Alcala_esp" channel="1" tbeg="195.150" dur="0.740" score="-24.118140" decision="YES"/>
</detected_termlist>
<detected_termlist termid="TEST-02" term_search_time="24.3" oov_term_count="0">
<term file="Ubeda_esp" channel="1" tbeg="82.110" dur="0.900" score="-69.872930" decision="YES"/>
<term file="Ubeda_esp" channel="1" tbeg="156.640" dur="0.750" score="-121.300726" decision="YES"/>
</detected_termlist>
<detected_termlist termid="TEST-03" term_search_time="24.3" oov_term_count="0">
<term file="Salamanca_esp" channel="1" tbeg="255.990" dur="0.640" score="-20.147835" decision="YES"/>
</detected_termlist>
<detected_termlist termid="TEST-04" term_search_time="24.3" oov_term_count="0">
</detected_termlist>
<detected_termlist termid="TEST-05" term_search_time="24.3" oov_term_count="0">
<term file="Baeza_esp" channel="1" tbeg="208.540" dur="0.720" score="-75.005138" decision="YES"/>
</detected_termlist>
```

Figura 3.20: Extracto de archivo STDLIST

Existe la posibilidad de cambiar un parámetro en el código que añade cierta inteligencia al sistema completo. Dicho parámetro es un umbral que desestimará las coincidencias con un score por debajo de él (más negativo) rellenando con la etiqueta 'NO' el parámetro 'decisión' del archivo STDLIST que este sistema devuelve como resultado. Por contra, mantendrá como válidas (decisión = YES) aquellas coincidencias cuyo score supere el umbral. Más adelante (en el capítulo 4) se verá cómo esta opción de variar el umbral del score es muy útil a la hora de mejorar los resultados de la evaluación.

Este subsistema tiene la funcionalidad extra de desechar las coincidencias repetidas debido al solapamiento temporal de los lattices. Si el sistema encuentra una coincidencia, ésta estará tanto en un lattice como en el siguiente o en el anterior (depende de si la coincidencia está en la primera mitad del tiempo al que se refiere el lattice o en la segunda). El conversor actúa de forma que si en el archivo MLF hay dos o más coincidencias con aproximadamente el mismo instante de inicio y de fin, únicamente la almacena una vez en el archivo STDLIST de cara a no obtener falsas alarmas espúreas en la evaluación de las búsquedas.

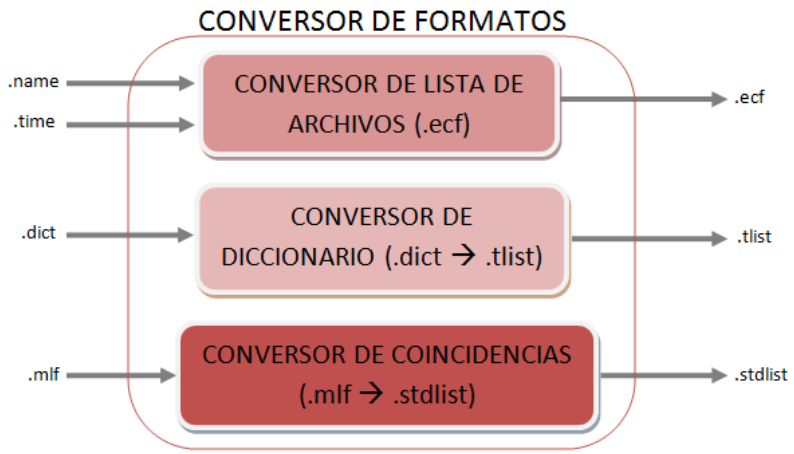

Figura 3.21: Estructura interna del conversor de formatos

3.3.2.3. Llamada al bloque

La llamada a este bloque tiene la siguiente forma:

./conversor ./Originales/diccionario.dict ./Originales/Nombres/nombres.name ./Originales/Tiempos/tiempos.time ./Originales/salida.mlf id

donde los campos son las entradas del bloque descritas anteriormente salvo el campo 'id' que es el nombre raíz (luego cada uno tendrá su extensión) que se le quiere dar a los archivos de salida del bloque. Es dentro del programa principal donde se realiza secuencialmente la llamada a los distintos subbloques de los que se compone el sistema.

3.3.3. Alineador de audio

La funcionalidad de este bloque es la de generar el alineamiento de los audios originales que será usado por la herramienta de evaluación de las búsquedas. Se puede decir que da como resultado lo que realmente hay en el audio y nos sirve para compararlo con lo que nuestro sistema dice que hay en el audio. Como se verá a continuación se compone de cuatro grandes bloques y aunque sea un complemento dentro de nuestro sistema completo, es una de las partes que más trabajo, esfuerzo y complejidad comprende.

3.3.3.1. Entradas y salidas del sistema

Las entradas del sistema son:

- Los archivos de audio (WAV) que contienen el sonido de los vídeos de información turística que conforman la base de datos descrita en la sección 3.2.1. En la Tabla 3.1 se puede ver una relación de todos ellos. En este caso, sólo se usan los archivos correspondientes al idioma español, como en el resto del sistema.
- Las transcripciones originales correspondientes a la voz contenida en los archivos de audio que se acaban de mencionar (únicamente los de idioma español) descritos en la sección 3.2.2. Están en formato TXT, se ven reflejadas en la Tabla 3.2 y se puede observar su estructura en la Figura 3.2.

La salida del sistema es:

- Un archivo en formato RTTM (Rich Transcription Time Mark) que contiene el alineamiento de cada una de las palabras que aparecen en el audio original. En este caso se ha creado un sólo archivo RTTM con todo el audio alineado ya que siempre se van a evaluar búsquedas realizadas en todo el audio. El formato de este archivo se contempla en la Figura 3.22 y en él, cada línea se refiere a una palabra o elemento del audio (pueden ser también silencios o música) con los siguientes campos:
  - Tipo de elemento: este campo puede tomar los valores 'LEXEME' en caso de ser una palabra o 'NON-LEX' en caso de ser un silencio o música.
  - Archivo de audio al que pertenece el elemento.
  - Canal del archivo de audio al que pertenece el elemento.
  - Instante de inicio del elemento dentro de su archivo de audio.
  - Duración del elemento.
  - Texto del elemento: en caso de ser una palabra es la propia palabra mientras que si es un silencio puede ser '_' cuando es una pausa entre palabras, 'R1' cuando es un silencio al principio de un tramo de voz o 'R2' cuando es un silencio al final de un tramo de voz. Los tramos de música se representan con un elemento 'R1' seguido de uno 'R2'.

- Subtipo de elemento: este campo puede tomar varios valores pero en nuestro caso sólo toma dos. Puede ser 'lex' si es una palabra u 'other' si es un silencio o música.
- Locutor correspondiente a ese elemento. En nuestro caso y dado que no nos importa el locutor este campo siempre vale 'LOC'.
- Un último campo cuyo valor siempre es nulo: '<NA>'.

```
NON-LEX Alcala_esp 1 26.453 0.54 R1 other LOC <NA>
LEXEME Alcala_esp 1 26.993 0.56 alcalá lex LOC <NA>
LEXEME Alcala_esp 1 27.553 0.5 de lex LOC <NA>
NON-LEX Alcala_esp 1 28.053 0.9 _ other LOC <NA>
LEXEME Alcala_esp 1 28.953 0.37 henares lex LOC <NA>
NON-LEX Alcala_esp 1 29.323 0.05 R2 other LOC <NA>
NON-LEX Alcala_esp 1 29.514 0.78 R1 other LOC <NA>
NON-LEX Alcala_esp 1 30.294 4.87 R2 other LOC <NA>
NON-LEX Alcala_esp 1 35.304 0.36 R1 other LOC <NA>
LEXEME Alcala_esp 1 35.664 0.52 situada lex LOC <NA>
LEXEME Alcala_esp 1 36.184 0.04 a lex LOC <NA>
LEXEME Alcala_esp 1 36.224 0.36 treinta lex LOC <NA>
LEXEME Alcala_esp 1 36.584 0.59 kilómetros lex LOC <NA>
LEXEME Alcala_esp 1 37.174 0.08 de lex LOC <NA>
LEXEME Alcala_esp 1 37.254 0.34 madrid lex LOC <NA>
```

Figura 3.22: Extracto de archivo RTTM

3.3.3.2. Estructura interna de bloques

Este sistema está internamente compuesto por varios bloques, que realizan una función distinta dentro del sistema y cada uno con sus entradas y salidas. La estructura interna del sistema se puede ver en la Figura 3.23. Los bloques de los que se compone se detallan a continuación.

Figura 3.23: Estructura interna del alineador de audio

Bloque segmentador de audio (2Segments)

Sus entradas son directamente los archivos WAV de los audios completos de la base de datos y las transcripciones originales (en TXT), es decir, las entradas del sistema de alineamiento completo, y un directorio donde dejar el resultado de la segmentación.

Sus salidas son una serie de archivos correspondientes a la segmentación del audio introducido:

- Archivos WAV con el audio correspondiente a cada segmento fruto de la segmentación.
- Archivos LAB con las etiquetas del texto contenido en ese segmento.
- Archivos SPK con el locutor del segmento correspondiente.

- Archivos CHN con el canal dentro del audio en el que se encuentra el segmento.
- Archivos STA con el instante de inicio del segmento dentro del audio completo.
- Archivos END con el instante de fin del segmento dentro del audio completo.

En la Figura 3.24 se puede ver un ejemplo de archivo LAB, SPK, STA y END (ya que WAV es audio y CHN en este caso no contiene nada).

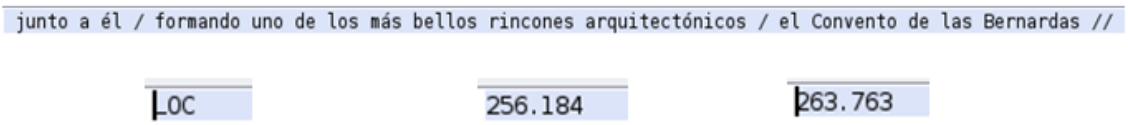

Figura 3.24: Ejemplos de archivos LAB (arriba), SPK, STA y END (abajo de izquierda a derecha)

Este bloque se compone de una serie de scripts escritos en distintos lenguajes de *ShellScripting* (tcsh, tcl) que cogen cada línea del cuerpo del archivo de la transcripción del audio y parten dicho audio según ese patrón de tiempos. Esto se hace con la precaución de no crear el doble de archivos debido a la aparición del texto dos veces dentro de la transcripción (como ya se vió, una para la prosodia y otra para la ortografía). La Figura 3.25 muestra el diagrama de caja negra de este bloque.

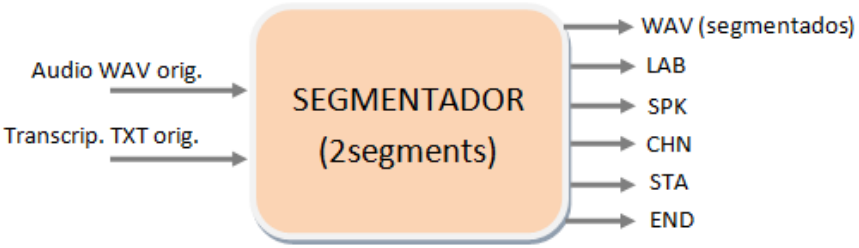

Figura 3.25: Caja negra del bloque segmentador de audio

Bloque Transcriptor (Transcriber)

Su entrada es el nombre del directorio donde se encuentran los archivos de salida del bloque anterior, de donde utilizará en particular los archivos LAB de etiquetas.

Sus salidas son varios archivos correspondientes a la transcripción en formato HTK de los segmentos que existen en el directorio de entrada:

- Archivos LABWORD en los que cada línea es la transcripción de una palabra dentro del segmento.
- Archivos LABSYL en los que cada línea es la transcripción de una sílaba dentro del segmento. Las palabras vienen separadas entre ellas por el símbolo '[_]' y durante la ejecución se puede observar la vocal acentuada con un símbolo '+' detrás (Figura 3.26).
- Archivos LABPHON en los que cada línea es la transcripción de un fonema dentro del segmento. Las sílabas se separan por un punto y las palabras se siguen separando mediante el símbolo '[_]'.

```
. d e+
. s u+ s
. n u e+ . b e
. p u e+ r . t a s
. n o+ s
. a . d e n . t r a+ . m o s
. e+ n
. u+ n
. b i a+ . x e
. p o+ r
. e+ l
. t i e+ m . p o
```

Figura 3.26: Ejemplo de la localización de la acentuación de las palabras durante la transcripción

En los tres archivos que se acaban de describir se incluyen los símbolos de principio (símbolo 'R1') y fin (símbolo 'R2') de segmento así como pausas cortas entre algunas palabras (símbolo '_'). En la Figura 3.27 aparecen ejemplos de estos tres tipos de archivos.

```
R1
junto
a
él
_
formando
uno
de
los
más
bellos
rincones
arquitectónicos
_
el
convento
de
las
bernardas
R2
```

```
R1
xun
to
[_]
a
[_]
el
_
for
man
do
[_]
u
no
[_]
de
[_]
los
[_]
mas
```

```
R1
.
x
u
n
.
t
o
[_]
.
a
[_]
.
e
l
_
.
f
o
r
```

Figura 3.27: Extractos de archivos LABWORD, LABSYL y LABPHON (de izq. a dcha.)

Este bloque se compone de algunos scripts escritos en distintos lenguajes (tcsh, perl, awk) que hacen uso de un transcriptor externo que es quien realiza la transcripción. La Figura 3.28 muestra la estructura de este bloque.

Figura 3.28: Caja negra del bloque transcriptor

Bloque Alineador (Aligner)

Sus entradas son tanto el directorio que contiene los archivos de salida de los dos bloques anteriores (de los que usará principalmente los segmentos de audio WAV y los archivos LABPHON, LABSYL y LABWORD) como los modelos acústicos que utiliza.

Sus salidas son unos archivos de etiquetas con formato HTK que contienen el alineamiento del audio por palabras (archivos LABWORDALI), sílabas (archivos LABSYLALI) y fonemas

(archivos LABPHONALI). En todos ellos en cada línea se ve reflejado el instante de inicio y fin en formato HTK y la palabra, sílaba o fonema en cada caso. En la Figura 3.29 aparecen ejemplos de estos tres tipos de archivos.

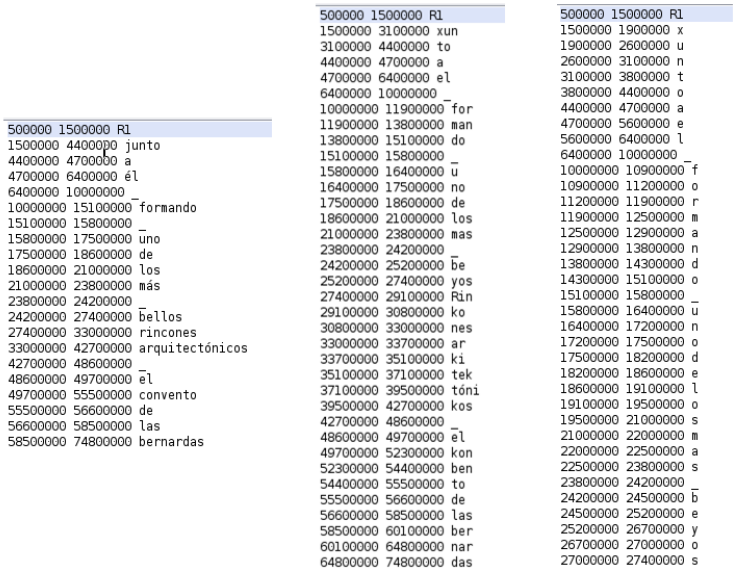

Figura 3.29: Extractos de archivos LABWORDALI, LABSYLALI y LAPHONALI (de izq. a dcha.)

Este bloque se compone de una serie de scripts escritos en distintos lenguajes (tcsh, perl, awk) que llaman a distintas herramientas de HTK:

- HHed: Sirve para manipular definiciones de Modelos Ocultos de Markov (HMMs). Básicamente los carga y modifica algunos parámetros.
- HParse: Sirve para generar lattice a nivel de palabra (para ser usados por la herramienta HVite) a partir de un texto con la descripción sintáctica.
- HVite: Esta herramienta es un reconocedor de palabras de propósito general de Viterbi. Hace coincidir un archivo de voz con una red de modelos HMM y devuelve una transcripción para cada archivo de voz.

Con estas herramientas y haciendo uso de los modelos fonéticos de español proporcionados por el tutor de este proyecto Doroteo Torre Toledano, se obtiene el alineamiento de palabras, sílabas y fonemas. Para ello, en una primera parte del bloque (en el script Aligner.tcsh) se realiza el alineamiento de fonemas a partir de los segmentos de audio WAV y de los LABPHON y a continuación (en el script AlignerSylWord.tcsh), mediante los propios LABPHON, los LABPHONALI obtenidos y los LABWORD y LABSYL, se consiguen los alineamientos de palabras y sílabas. La Figura 3.30 muestra la estructura interna de este bloque.

Para la aplicación en la que se centra este proyecto, los alineamientos de sílabas no son útiles por lo que no se usarán.

Bloque conversor de audio alineado HTK a RTTM (MAVIR2RTTM)

Sus entradas son las transcripciones originales (TXT) y los archivos LABWORDALI que se generan en el bloque anterior que se encuentran en el directorio de las salidas de todos los bloques anteriores.

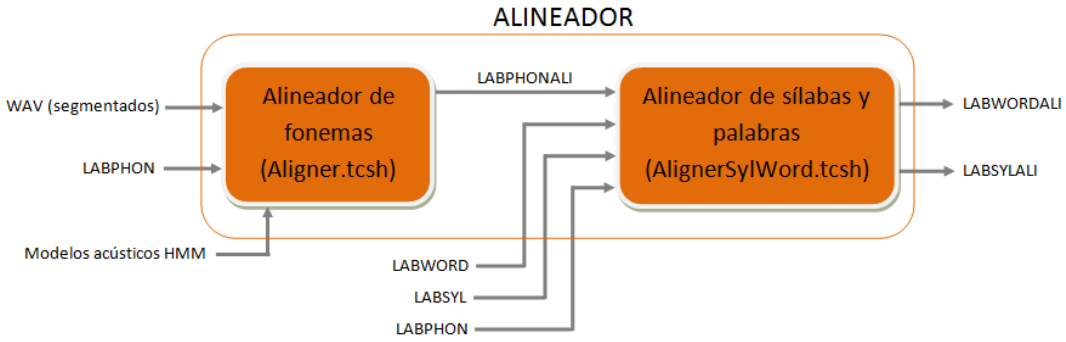

Figura 3.30: Estructura interna del bloque alineador

Su salida es un archivo RTTM (cuyo formato se observa en la Figura 3.22) por cada archivo de audio original, es decir, por cada directorio que exista en el directorio pasado como entrada.

Este bloque se compone de unos scripts que lo único que hacen es una conversión del formato HTK devuelto por el bloque anterior al formato RTTM que requiere el evaluador de búsquedas y que se ha descrito con anterioridad. En concreto en este proyecto al final de este bloque se ha procedido a juntar la salida de todos los RTTM en uno sólo de cara a la evaluación de las búsquedas conjuntas en todo el audio disponible. La Figura 3.31 muestra la estructura interna de este bloque.

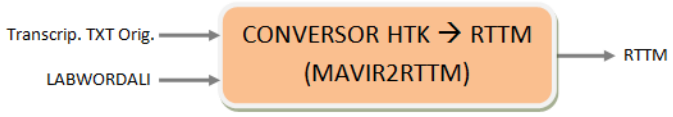

Figura 3.31: Caja negra del bloque conversor a RTTM

Con este sistema de alineamiento de audio ha habido muchas complicaciones debido sobretodo a problemas con las incompatibilidades en la codificación de texto a la hora de escribir tildes (es por ello que los archivos de sílabas son erróneos en parte, cosa que no importa para este proyecto ya que no se usan).

3.3.3.3. Llamada al bloque

Debido a que el sistema se compone de varios bloques como se ha explicado en la sección anterior, la llamada al sistema se compone de varias subllamadas, quedando de la siguiente forma para cada archivo de audio original a alinear:

cd 2segments
./2segments.tcsh $FichWav $FichTranscrip $DirOut
cd ../Transcriber
./Transcriber.tcsh $DirOut
cd ../Aligner
./Aligner.tcsh $DirOut
./AlignerSylWord.tcsh $DirOut
cd ../MAVIR2RTTM
./MAVIR2RTTM.tcl $LabWordAli $DirOut $RTTMFile

Finalmente se juntan todos los RTTM en uno sólo.

3.3.4. Evaluador de búsquedas realizadas con Spoken Term Detection (STD)

Este sistema se compone básicamente de la herramienta de evaluación de STD desarrollada por el NIST, de la que ya se han analizado los detalles internos en la sección anterior. Se encarga de determinar cómo de buena es una búsqueda de términos realizada.

3.3.4.1. Entradas y salidas del sistema

Las entradas de este sistema son las siguientes:

- Un archivo ECF de lista de archivos de audio que forma parte de la salida del bloque conversor explicado en uno de los apartados anteriores. Un ejemplo de sus estructura está en la Figura 3.18.
- Un archivo TLIST de lista de términos de búsqueda que también forma parte de la salida del bloque conversor de formatos. Se puede ver un ejemplo de sus estructura en la Figura 3.19.
- Un archivo STDLIST de coincidencias que se obtiene tras la ejecución del bloque conversor de formatos explicado en uno de los apartados anteriores. Un ejemplo de sus estructura se encuentra en la Figura 3.20.
- Un archivo RTTM que contiene el alineamiento de todo el audio que se ha usado en la búsqueda. Este archivo se obtiene a partir de los audios y las transcripciones originales y su proceso de obtención se ha detallado en el apartado anterior donde se analiza el bloque diseñado a tal efecto. Un ejemplo de su estructura aparece en la Figura 3.22.

Las salidas de este sistema son las siguientes:

- Un archivo con extensión OCC.TXT en el que se recogen todas las estadísticas de la búsqueda evaluada. En él se ven tanto los resultados de la evaluación particularizados para cada uno de los términos buscados como los resultados globales. En la Figura 3.32 se refleja un ejemplo de este tipo de archivo. Su estructura es la siguiente:

  1. Cabecera: En ella se muestran el idioma, el identificador e información del score entre otros parámetros.
  2. Cuerpo: En él se proporciona información sobre las estadísticas de cada término buscado. Además del identificador del término y su texto se detallan los siguientes campos:
     - Ref: Es el número de coincidencias de ese término que existen realmente en el audio. Esta información se extrae del archivo RTTM en el que se encuentra el alineamiento de dicho audio.
     - Corr: Es el número de coincidencias correctas de ese término de entre las encontradas en los lattices en la búsqueda realizada mediante STD.
     - FA: Es el número de Falsas Aceptaciones (en inglés False Alarms) de ese término encontradas en los lattices, es decir, las coincidencias encontradas mediante STD que no coinciden con lo que existe realmente en el audio.
     - Miss: Es el número de Falsos Rechazos de ese término, es decir, las veces que ese término aparece en el audio realmente y nuestro sistema de STD las ha pasado por alto, las pérdidas (en inglés Misses) de ese término que hemos cometido.

- Occ. Value: Es el Occurence-Weighted Value (OCC) particularizado para ese término. Como se vió en el capítulo 2, el OCC se rige por la fórmula:

$$OCC = \frac{\sum_{terms}[VN_{correct}(t) - C_{FA}N_{FA}(t)]}{\sum_{terms} VN_{true}(t)}$$

  - t es cada término del diccionario de búsqueda.
  - $N_{correct}(t)$ es el número de coincidencias encontradas del término t que son correctas.
  - $N_{FA}(t)$ es el número de falsas alarmas del término t.
  - $N_{true}(t)$ es el número real de veces que el término t aparece en el audio.
  - V es el peso que se le asigna a las coincidencias correctas.
  - C_{FA} es el peso que se le asigna a las falsas alarmas.

  Este valor representa una medida de la calidad de la búsqueda ya que tiene en cuenta las falsas alarmas y las coincidencias correctas del término estudiado.
- P(FA): Es la frecuencia de ocurrencia de una Falsa Aceptación de ese término expresado en veces por segundo. Teóricamente este valor no es una probabilidad ya que podría ser mayor que la unidad.
- P(Miss): Es la probabilidad de que se produzca una pérdida de una coincidencia de ese término.

3. Resumen: En él se recogen las sumas de los valores de estadísticas que se han mencionado para cada término en particular y la media de los mismos, en definitiva, los valores referidos a la búsqueda completa. Entre ellos se puede observar el valor más importante que vamos a tener en cuenta a la hora de decidir si el resultado de la búsqueda es mejor o peor, el ATWV (Actual Term-Weighted Value) que sigue la fórmula presentada a continuación:

$$ATWV = 1 - \frac{\sum_{terms}[P_{Miss}(t) + \beta P_{FA}(t)]}{T}$$

   - t es cada término del diccionario de búsqueda.
   - $P_{Miss}(t)$ es la probabilidad de pérdida del término t.
   - $P_{FA}(t)$ es la probabilidad de falsa alarma del término t.
   - T es el número total de términos que hay en el diccionario de búsqueda.
   - $\beta = \frac{C}{V}(P_{prior}^{-1}(t) - 1)$
     - C y V son los pesos que se les asigna a las falsas alarmas y a las coincidencias correctas respectivamente. $\frac{C}{V}$ en la evaluación del NIST vale 0.1 ya que se le da más importancia (más peso) a las coincidencias correctas.
     - $P_{prior}(t)$ en la evaluación del NIST es 10^{-4}
     - Con estos valores, β vale aproximadamente 1000.

   Finalmente se puede ver un resumen de la evaluación de la búsqueda de cara a la representación de la curva DET.

Todos los parámetros descritos anteriormente que definen la evaluación están relacionados entre sí de la siguiente forma:

$$Corr + Miss = Ref$$

$$Corr + FA = C_{tot}$$

$$P(FA) = \frac{FA}{T_{tot}}$$

$$P(Miss) = \frac{Miss}{Ref}$$

donde C_{tot} son las coincidencias totales encontradas por el sistema STD y T_{tot} es el tiempo total de audio, el que suman entre todos los audios analizados.

```
+-------------------------------------------------------------------------------+
| Indexing Time:                1.0000                                          |
| Language:                    spanish                                          |
| Index size (bytes):                1                                        |
| System ID:                     MAVIR                                         |
| Coefficient C:                 0.1000                                        |
| Coefficient V:                 1.0000                                        |
| Trials Per Second:             1.0000                                        |
| Probability of a Term:         0.0001                                        |
| Decision Score          OK (Max NO: -230.8077, Min YES: -214.0325)                                  |
+---------------------------------------+---------------------------------------+
|                                  Search |                   ALL                 |
|                                         |                  Occ.                 |
|    TermID                  Text    Time |    Ref  Corr   FA  Miss  Value P(FA)  P(Mis)|
+---------------------------------------+---------------------------------------+
| TEST-01                universidad    24.30 |     9     3     0     6 0.333 0.00000 0.667 |
| TEST-02               arquitectura    24.30 |    15     1     0    14 0.067 0.00000 0.933 |
| TEST-03               ayuntamiento    24.30 |     4     1     0     3 0.250 0.00000 0.750 |
| TEST-04                  barcelona    24.30 |     4     2     2     2 0.450 0.00031 0.500 |
| TEST-05                  modernismo   24.30 |     2     2     1     0 0.950 0.00016 0.000 |
| TEST-06                  plateresca   24.30 |     3     0     0     3 0.000 0.00000 1.000 |
| TEST-07                  compostela   24.30 |     7     0     1     7 -0.014 0.00016 1.000 |
| TEST-08                  guadalajara  24.30 |     2     0     0     2 0.000 0.00000 1.000 |
| TEST-09                  cristianos   24.30 |     5     1     0     4 0.200 0.00000 0.800 |
| TEST-10                  delirante    24.30 |     1     0     3     1 -0.300 0.00047 1.000 |
| TEST-11                  guadarrama   24.30 |     2     1     2     1 0.400 0.00031 0.500 |
| TEST-12                  extremadura  24.30 |     2     0     0     2 0.000 0.00000 1.000 |
| TEST-13                  cantÃibrico  24.30 |     1     0     1     1 -0.100 0.00016 1.000 |
| TEST-14                mediterrÃineo  24.30 |     1     1     1     0 0.900 0.00016 0.000 |
| TEST-15                      tenerife   24.30 |     1     0     0     1 0.000 0.00000 1.000 |
| TEST-16                  cervantino   24.30 |     1     0     0     1 0.000 0.00000 1.000 |
| TEST-17                  catalunya    24.30 |     1     0     0     1 0.000 0.00000 1.000 |
| TEST-18                albarracÃ-n   24.30 |     1     0     0     1 0.000 0.00000 1.000 |
| TEST-19                  salamanca    24.30 |     3     2     0     1 0.667 0.00000 0.333 |
| TEST-20                  jerusalÃ©n   24.30 |     2     0     0     2 0.000 0.00000 1.000 |
| TEST-21                  acueducto    24.30 |     1     0     0     1 0.000 0.00000 1.000 |
| TEST-22                  isabelino    24.30 |     2     2     1     0 0.950 0.00016 0.000 |
| TEST-23                agricultura    24.30 |     1     0     1     1 -0.100 0.00016 1.000 |
| TEST-24                  velÃizquez   24.30 |     2     0     2     2 -0.100 0.00031 1.000 |
+---------------------------------------+---------------------------------------+
| Totals, Actual Occ. Weighted Value    583.20 |    73    16    15    57 0.199 0.00237 0.781 |
| Means (N/A excl.)                      24.30 |     3     0     0     2 0.190 0.00010 0.770 |
+---------------------------------------+---------------------------------------+
|                                       |  Number of Trials                6399     |
|                                       |  Total Speech Time (sec.)        6399.0   |
|                                       |  A. T-Weighted Value (N/A excl.)     0.132  |
+---------------------------------------+---------------------------------------+

+-------------------------------------------------------------+
|                  DET Curve Analysis Summary                 |
+-------------+---------------------+-------------------------+
|             |      Weighted       |                Decision |
| Description | Max Value | A. Value |   P(Fa) P(Miss)      Score |
+-------------+-----------+----------+-------------------------+
+-------------+-----------+----------+-------------------------+
|         ALL |   0.1647  |  0.1322  | 0.00007   0.770   -194.57210700 |
```

Figura 3.32: Ejemplo de archivo OCC

- Un archivo con extensión ALI.TXT en el que se reproduce para cada término, todas las coincidencias que se producen en cada uno de los archivos (sólo en los que se haya producido alguna). De esta forma, aparecen tanto las coincidencias reales que existen en el audio como las encontradas por el sistema STD que en muchos casos se solaparán, en concreto, en los casos en los que el sistema haya funcionado correctamente. En la Figura 3.33 se puede ver un ejemplo de la estructura de este archivo. Cada entrada del archivo ALI se compone de dos columnas con varios campos que se describen a continuación:

  1. Ref: Columna con los datos relativos a las coincidencias reales existentes en el audio:

- BT (Beginning Time): Instante de inicio de la coincidencia dentro de ese archivo de audio.
- ET (Ending Time): Instante de fin de la coincidencia dentro de ese archivo de audio.

2. Sys: Columna con los datos relativos a las coincidencias encontradas por el sistema de búsqueda STD:
   - BT: Instante de inicio de la coincidencia.
   - ET: Instante de fin de la coincidencia.
   - Score: Puntuación asignada a esa coincidencia (valor negativo).
   - Dec.: Decisión tomada sobre si esa coincidencia es válida o no (puede ser YES o NO).

```
TERM: universidad
    FILE: Alcala_esp
        CHANNEL: 1
        +----------------------+----------------------------------------+
        |         Ref          |                  Sys                   |
        |       BT        ET   |       BT        ET      Score     Dec. |
        +----------------------+----------------------------------------+
        |     86.687     87.467 |     86.690     87.44 -57.888176    YES  |
        |    121.645    122.295 |        --         --         --     --   |
        |    152.691    153.111 |        --         --         --     --   |
        |    163.559    164.179 |    163.560     164.21 -108.467311    YES  |
        |    195.146    195.856 |    195.150     195.89 -24.118140    YES  |
        +----------------------+----------------------------------------+
    FILE: Baeza_esp
        CHANNEL: 1
        +----------------------+----------------------------------------+
        |         Ref          |                  Sys                   |
        |       BT        ET   |       BT        ET      Score     Dec. |
        +----------------------+----------------------------------------+
        |    148.582    149.172 |        --         --         --     --   |
        |    214.831    215.561 |        --         --         --     --   |
        +----------------------+----------------------------------------+
    FILE: Salamanca_esp
        CHANNEL: 1
        +----------------------+----------------------------------------+
        |         Ref          |                  Sys                   |
        |       BT        ET   |       BT        ET      Score     Dec. |
        +----------------------+----------------------------------------+
        |     40.413     41.103 |        --         --         --     --   |
        |     75.172     75.832 |        --         --         --     --   |
        +----------------------+----------------------------------------+

TERM: arquitectura
    FILE: Aranjuez_esp
        CHANNEL: 1
        +----------------------+----------------------------------------+
        |         Ref          |                  Sys                   |
        |       BT        ET   |       BT        ET      Score     Dec. |
        +----------------------+----------------------------------------+
        |     37.929     38.599 |        --         --         --     --   |
        +----------------------+----------------------------------------+
    FILE: Baeza_esp
        CHANNEL: 1
        +----------------------+----------------------------------------+
        |         Ref          |                  Sys                   |
        |       BT        ET   |       BT        ET      Score     Dec. |
        +----------------------+----------------------------------------+
        |    226.962    227.632 |        --         --         --     --   |
        +----------------------+----------------------------------------+
```

Figura 3.33: Extracto de archivo ALI

- Un archivo con extensión DET.PNG que es una imagen con la curva DET correspondiente a la búsqueda. Una curva DET es una representación de las falsas alarmas frente a los falsos rechazos o pérdidas, como se vio en el capítulo 2. Un ejemplo de las curvas DET que devuelve el sistema de evaluación se puede ver en la Figura 3.34.
- Un archivo con extensión CACHE que contiene, término a término, las coincidencias encontradas en el archivo RTTM de alineamiento, es decir, las coincidencias de los términos a buscar que realmente existen en los audios originales. Una muestra de su estructura se muestra en la Figura 3.35. Dicha estructura sigue el siguiente patrón:

  1. Cabecera con información sobre el archivo RTTM.

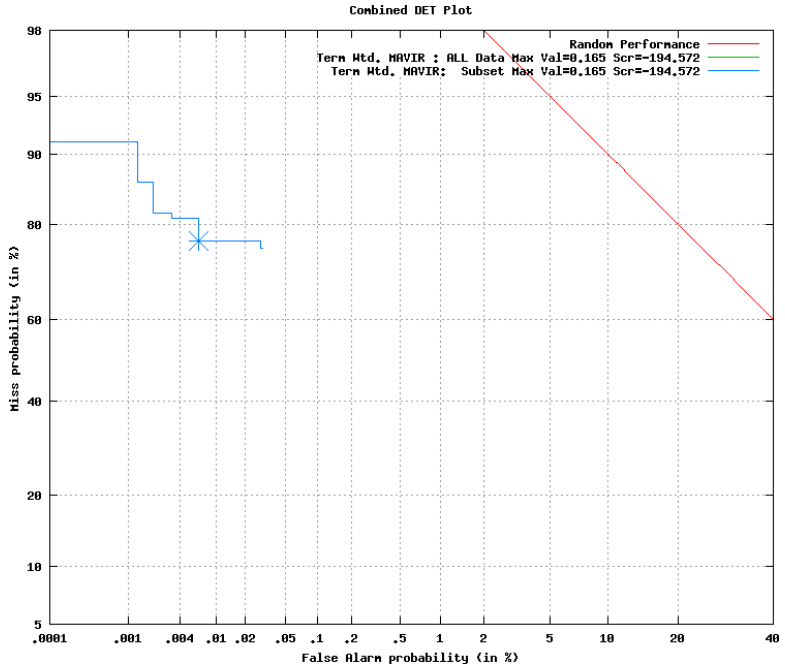

Figura 3.34: Ejemplo de curva DET

2. Cuerpo en el que cada entrada se refiere a un término y tiene la siguiente estructura:
   - Una línea denotada por la partícula <term> con el identificador y el texto del término.
   - Una o varias líneas denotadas por la partícula <occurrence> con las coincidencias de ese término. Se detallan: archivo donde se halla la coincidencia, canal, instante de inicio y duración de la misma.

```
<rttm_cache_file system_V="1751439" find_threshold="0.5">
    <term termid="TEST-19"><termtext>salamanca<termtext>
        <occurrence file="Salamanca_esp" channel="1" begt="28.486" dur="1.3900"/>
        <occurrence file="Salamanca_esp" channel="1" begt="38.163" dur="0.5500"/>
        <occurrence file="Salamanca_esp" channel="1" begt="298.57" dur="0.6400"/>
    </term>
    <term termid="TEST-20"><termtext>jerusalén<termtext>
        <occurrence file="SantiagoCompostela_esp" channel="1" begt="310.642" dur="0.6500"/>
        <occurrence file="Segovia_esp" channel="1" begt="189.644" dur="0.6000"/>
    </term>
    <term termid="TEST-24"><termtext>velázquez<termtext>
        <occurrence file="Madrid_esp" channel="1" begt="27.126" dur="0.5600"/>
        <occurrence file="Valencia_esp" channel="1" begt="146.795" dur="0.5300"/>
    </term>
    <term termid="TEST-13"><termtext>cantábrico<termtext>
        <occurrence file="Gaudi_esp" channel="1" begt="102.646" dur="0.6600"/>
    </term>
    <term termid="TEST-23"><termtext>agricultura<termtext>
        <occurrence file="Ubeda_esp" channel="1" begt="43.163" dur="0.4700"/>
    </term>
```

Figura 3.35: Extracto de archivo CACHE

- Otros archivos relativos principalmente al archivo DET.PNG anteriormente citado. Son archivos destinados a representar la curva DET y otras curvas complementarias a ella. Algunos contienen el código de GNUPLOT para obtener la gráfica (extensión PLT) y otros los propios datos con los que se dibuja la curva (extensión DAT). También hay imágenes que complementan a la curva DET (extensión PNG). En la Figura 3.36, Figura3. 3.37 y Figura 3.38 se pueden ver ejemplos de cada uno de estos tipos de archivos.

La caja negra correspondiente al sistema evaluador se puede observar en la Figura 3.39.

```
## GNUPLOT command file
set terminal postscript color
set data style lines
set noxzeroaxis
set noyzeroaxis
set key top spacing .5
set size ratio 0.821894871074622
set noxtics
set noytics
set title 'Combined DET Plot'
set ylabel 'Miss probability (in %)'
set xlabel 'False Alarm probability (in %)'
set grid
set pointsize 3
set ytics (\
    '5' -1.6449, '10' -1.2816, '20' -0.8416, '40' -0.2533, '60' 0.2533, \
    '80' 0.8416, '90' 1.2816, '95' 1.6449, '98' 2.0537)
set xtics (\
    '.0001' -4.7534, '.001' -4.2649, '.004' -3.9444, '.01' -3.7190, '.02' -3.5401, \
    '.05' -3.2905, '.1' -3.0902, '.2' -2.8782, '.5' -2.5758, '1' -2.3263, \
    '2' -2.0537, '5' -1.6449, '10' -1.2816, '20' -0.8416, '40' -0.2533)
plot [-4.75343910607888:-0.253347103317183] [-1.64485362793551:2.05374890849825] \
   -x title 'Random Performance' with lines 1,\
  '/home/pablo/Evaluaciones/30_largas/30_largas.det.sub00.dat.1' using 3:2 title 'Term Wtd. MAVIR : ALL Data Max Val=0.165 Scr=-194.572' with
lines 2,\
  '/home/pablo/Evaluaciones/30_largas/30_largas.det.sub00.dat.2' using 6:5 notitle with points 2,\
  '/home/pablo/Evaluaciones/30_largas/30_largas.det.sub01.dat.1' using 3:2 title 'Term Wtd. MAVIR:  Subset Max Val=0.165 Scr=-194.572' with lines
3,\
  '/home/pablo/Evaluaciones/30_largas/30_largas.det.sub01.dat.2' using 6:5 notitle with points 3
```

Figura 3.36: Extracto de archivo PLT

```
# DET Graph made by DETCurve
# PooledTotalTrials = 6399
# DET Type: Term Wtd.
# Abbreviations: ssd() is the sample Standard Deviation of a Variable
#                ppndf() is the normal deviant of a probability. ppndf(.5)=0
#                -2SE(v) is v - 2(StandardError(v)) = v - 2 * (sampleStandardDev / sqrt(n))
# score ppndf(Pmiss) ppndf(Pfa) Pmiss Pfa Value ppndf(-2SE(Pmiss)) ppndf(-2SE(Pfa)) ppndf(+2SE(Pmiss)) ppndf(+2SE(Pfa)) SE(Value)
-727.806258 0.69429057556794 -3.4261226270579 0.75625 0.000306132073448534 -0.0623514602411888 0.249400221132799 -4.05321463821083
1.36596870873596 -3.24512053055566 -0.290814389126183
-705.312711 0.69429057556794 -3.43195954275556 0.75625 0.000299618603592872 -0.0558386417325128 0.249400221132799 -4.00853971669271
1.36596870873596 -3.25413590543867 -0.280534558493891
-561.158777 0.69429057556794 -3.43791577929192 0.75625 0.000293105133737211 -0.0493258232238369 0.249400221132799 -3.97164382423412
1.36596870873596 -3.26334817870227 -0.270517899626393
-533.000923 0.69429057556794 -3.44399653737327 0.75625 0.000286591663881549 -0.04281300471516l 0.249400221132799 -4.01566066602221
1.36596870873596 -3.26696098357837 -0.273369420010208
-506.876375 0.739304328120453 -3.44399653737327 0.770138888888889 0.000286591663881549 -0.0567018936040498 0.311262525088389 -4.01566066602221
1.3922609581056 -3.26696098357837 -0.2680641419468
-447.239539 0.739304328120453 -3.45020736951899 0.770138888888889 0.000280078194025888 -0.0501890750953739 0.311262525088389 -3.97856249334717
1.3922609581056 -3.27651185445141 -0.258282110304509
```

Figura 3.37: Extracto de archivo DAT

3.3.4.2. Llamada al bloque

La llamada al programa de evaluación de búsquedas STDEval-0.7 tiene la siguiente forma:

perl -I ../STDEval-0.7/src/STDEval.pl -e ./archivos.ecf.xml -r ./todo.rttm -s ./coincidencias.stdlist.xml -t ./diccionario.tlist.xml -A -o ./estadisticas.occ.txt -a ./alineamiento_ coincidencias.ali.txt -d ./curva.det -c ./coincidencias_ reales.cache

donde se indican dónde están los archivos de entrada y se les da nombre a los que serán los archivos de salida con las distintas opciones:

- -e: Archivo ECF
- -r: Archivo RTTM
- -s: Archivo STDLIST
- -t: Archivo TLIST
- -o: Archivo OCC
- -a: Archivo ALI
- -d: Archivo DET
- -c: Archivo CACHE
- Además, se usa la opción -A (–All-display) que añade información sobre cada uno de los términos en el archivo de estadísticas OCC.

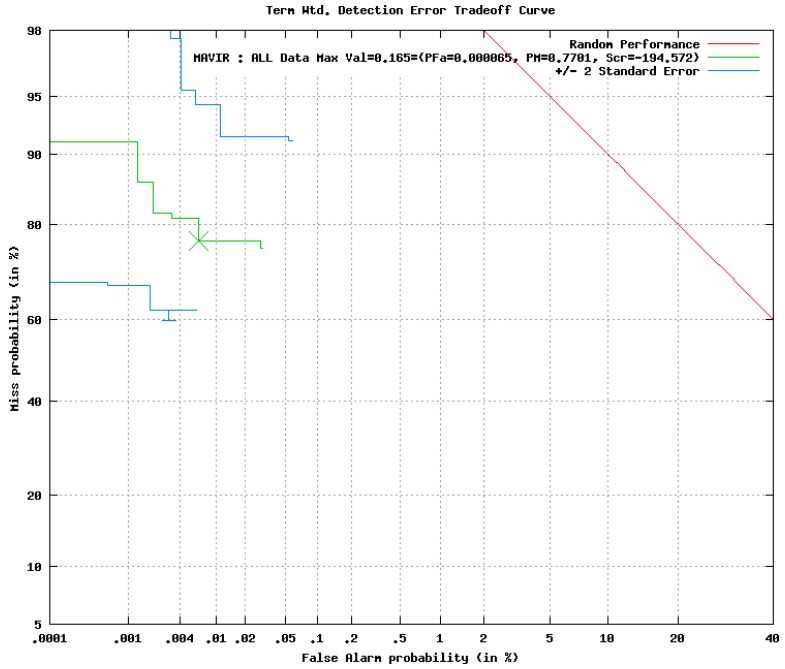

Figura 3.38: Ejemplo de archivo DET complementario a la curva DET

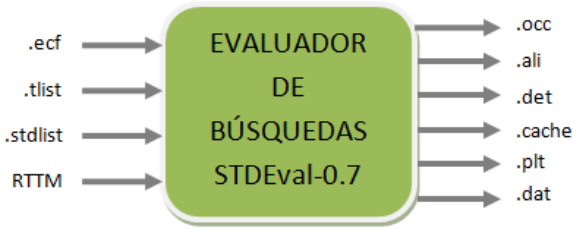

Figura 3.39: Estructura del sistema evaluador de búsquedas

En resumen, lo que hace básicamente este sistema es:

1. Generar el archivo CACHE buscando los términos en el alineamiento de todo el audio (archivo RTTM).
2. A partir del archivo CACHE (coincidencias reales) y del STDLIST (coincidencias encontradas por el sistema STD), genera el archivo ALI donde se consiguen alinear unas y otras coincidencias en los casos que proceda.
3. Analizando el archivo ALI, pasa a obtener las estadísticas de la búsqueda (archivo OCC).
4. Con dichas estadísticas se crea la gráfica de la curva DET correspondiente a la búsqueda y por tanto todos los archivos que giran en torno a dicha gráfica (PLT, DAT y PNG).

3.3.5. Sistema completo

Tras la explicación de cada bloque por separado sólo queda resumir en un diagrama toda la interconexión del sistema entero con las entradas y salidas de cada bloque, como se muestra en la Figura 3.40. De cara a la automatización de la ejecución del sistema completo de ha desarrollado

un script de *ShellScripting* de Linux (.tcsh) en el cual se va llamando a los distintos bloques del sistema con las llamadas vistas en los apartados anteriores.

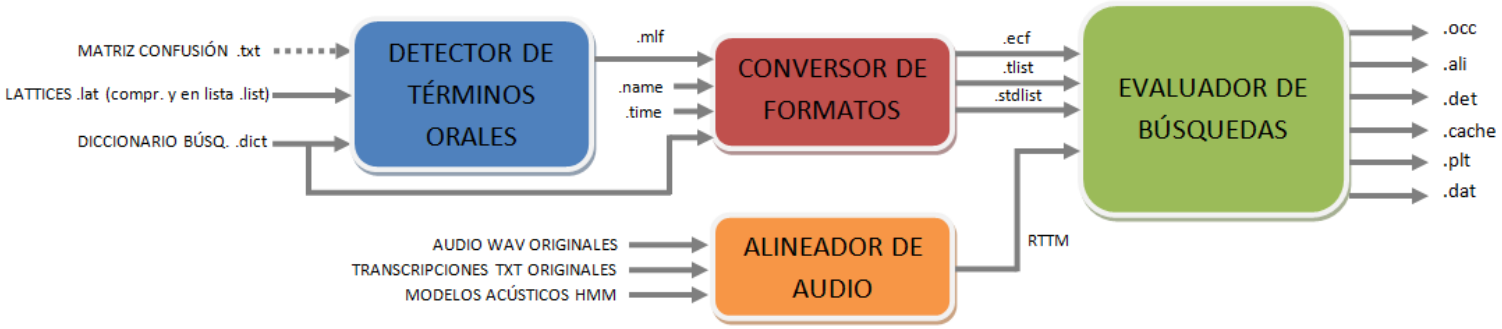

Figura 3.40: Estructura del sistema completo desarrollado con todas las entradas y salidas de cada uno de los subsistemas

4

Experimentos y resultados

4.1. Introducción

De cara a realizar una prueba integral del sistema desarrollado que se ha descrito minuciosamente en el capítulo anterior, se ha configurado un banco de pruebas lo más completo posible para poder observar y comparar cómo se comporta el sistema en distintas situaciones. De esa manera se podrá dar una valoración objetiva del rendimiento del sistema desarrollado y distinguir las condiciones para las que dicho sistema funciona mejor o peor.

Como se ha visto en el capítulo 3, las entradas a la parte del sistema que se ha desarrollado en este proyecto son:

- Lattices de fonemas fruto del reconocimiento de voz previo, simplificados para facilitar su compatibilidad con las transcripciones fonéticas de los términos de búsqueda y de esa forma no complicar la eleboración de los mismos. Estos lattices vienen dados por el reconocimiento, no se pueden modificar.
- Diccionario de búsqueda compuesto por los términos que se desean encontrar en el audio. En él se detallan tanto las palabras como su transcripción fonética. Es el usuario quien configura el diccionario.
- Opcionalmente una matriz de confusión en la que se ven las probabilidades de inserción, borrado y sustitución de cada uno de los fonemas. Viene impuesta por el idioma (en este caso el español) y por el estudio realizado a lo largo de los años sobre dicho idioma que dio lugar a la matriz.

Se puede apreciar cómo el único elemento de entrada al sistema que se puede modificar a antojo del usuario es el diccionario de búsqueda ya que si hablamos de los lattices, pudiendo usar siempre todo el conjunto de los mismos, no tiene sentido realizar pruebas parciales con los lattices correspondientes a unos o a otros archivos de audio. Estrictamente hablando, también se puede modificar el hecho de introducir o no una matriz de confusión para la búsqueda, experimento que se ha llevado a cabo.

Por lo tanto, el parámetro fundamental con el que se ha 'jugado' de cara a confeccionar el banco de pruebas ha sido el diccionario de búsqueda. Como se verá, los dos principales factores

del diccionario que se han variado han sido la longitud de sus palabras y el tamaño del propio diccionario.

Como se comentó en el capítulo anterior, el sistema cuenta con cierta inteligencia en la fase de conversión de formatos, existiendo un sistema decisor que en base al score de cada coincidencia da por buena o por errónea la misma. Pues bien, el score umbral que usa el decisor ha sido otro de los parámetros que se han variado y que da lugar a más experimentos del banco de pruebas que se detalla en la siguiente sección.

4.2. Experimentos realizados

A continuación se describe la totalidad de experimentos realizados detallando en cada uno todos los parámetros usados, tanto los que se han mantenido fijos como los que se han variado para observar el comportamiento del sistema en función de ellos. Existe un elemento de entrada al sistema que se ha mantenido invariable a lo largo de los experimentos y es que se ha introducido un archivo LIST con el nombre de todos los archivos de lattices disponibles, es decir, todos los experimentos se han realizado sobre todos los lattices y por tanto sobre todos los archivos de audio de la base de datos.

4.2.1. Experimento 1: Comparación de las distintas longitudes de palabras

En primer lugar, se ha realizado un experimento en el que se crean tres diccionarios de búsqueda con palabras de distintas longitudes, de cara a comprobar la variación en el comportamiento del sistema según el tamaño de las palabras buscadas. Estos diccionarios han sido creados a mano analizando en detalle las transcripciones originales para escribir términos que se encuentren en el audio. Además, se ha intentado que se trate de términos que sean nombres propios o acrónimos que son para los que STD está destinado y para los que se comporta mejor. Para observar únicamente cómo varían los resultados respecto a la longitud de las palabras se han mantenido constantes el resto de los parámetros que se pueden variar, es decir, los tres diccionarios creados son de 90 palabras y se ha usado un margen para el score de -250. Los diccionarios creados son:

- Diccionario de palabras largas: una lista de palabras de tres sílabas o más de longitud.
- Diccionario de palabras cortas: una lista de palabras de tres sílabas o menos de longitud.
- Diccionario de palabras mixtas: una lista con palabras de todas las longitudes creada a partir de palabras de los dos diccionarios anteriores (la mitad largas y la otra mitad cortas).

Estos diccionarios son la base del resto de diccionarios usados en el resto de experimentos. Es por ello que pueden ser observados en el Anexo A.

4.2.2. Experimento 2: Ajuste del umbral de score

En el segundo experimento se ha ajustado el umbral del score hasta conseguir los resultados óptimos para el equilibrio entre falsas alarmas y pérdidas y, por tanto, para el mejor valor ATWV. El umbral se ha variado gracias a la inteligencia del sistema introducida por el conversor de formatos que cuenta con la opción de configurar este parámetro de decisión.

Se ha calculado el score óptimo para palabras largas y se ha usado un diccionario de 90 palabras como el del experimento anterior. Se ha usado el método de prueba y error y los casos de umbral probados más representativos son:

- Sin umbral de score
- Umbral de score = -250
- Umbral de score = -225

Se ha procedido de igual forma para palabras mixtas y palabras cortas pero el experimento se centra en las palabras largas ya que no procede repetir los resultados tres veces cuando lo que se persigue es ver el efecto de encontrar un umbral óptimo y no la diferencia entre longitudes de palabra, ya vista en el experimento anterior. Es más, la tendencia de los resultados será similar comparando las distintas longitudes de términos.

4.2.3. Experimento 3: Comparación del tamaño de diccionario

El siguiente experimento ha consistido en comparar los distintos tamaños del diccionario. Para ello se ha utilizado un diccionario de 90 palabras como el de los experimentos anteriores y otro de 30 palabras (conseguidas eligiendo sin ningún criterio en particular palabras procedentes del diccionario de 90 palabras). Esto se ha hecho para cubrir todo el espectro de posibilidades a la hora de probar el sistema y ver qué ocurre, pero en realidad carece de mucho sentido ya que, en principio, cuanto más grande sea el diccionario usado, más representativos serán los resultados. En este experimento se usan palabras largas y se fija el valor del umbral de score en -225, para dejar constantes el resto de factores y de esa forma observar sólo la influencia del tamaño del diccionario.

4.2.4. Experimento 4: Análisis de la influencia de los términos en plural y de los términos que pueden aparecer dentro de otros términos

En el experimento número 4 se ha querido comprobar y cuantificar el impacto que tienen algunos tipos de términos cuya naturaleza provoca el deterioro de los resultados. Para ello se han modificado los diccionarios de forma que no cuenten con dichos tipos de palabras. Los diccionarios de búsqueda originales (antes de eliminar términos) eran los usados en el primer experimento y los diccionarios resultantes de esta 'criba' se pueden ver en el Anexo A junto con el resto de diccionarios usados. Los dos principales tipos de términos de los que se va a estudiar la influencia que tienen en el resultado de la evaluación son:

- Los términos en plural, ya que un suceso frecuente es el hecho de que aparezca en el audio la palabra en singular y no sea encontrada porque el usuario haya incluido en el diccionario de búsqueda la palabra en plural (que también se encuentra en el audio).
- Palabras muy fácilmente confundibles con otras o que forman parte de alguna otra palabra más larga (como 'maría' que se confunde muy fácilmente con otros términos como 'manía' o 'masía' y forma parte de términos como 'amaría' o 'quemaría').
    - Algunos de ellos, además, son términos que no son nombres propios o acrónimos. Por ejemplo, 'Barcelona' al ser un nombre propio y por tanto un término más peculiar, no es fácil que se confunda con otros términos, mientras que 'casa' al no tener esa cierta peculiaridad de los nombres propios se puede confundir fácilmente (con 'masa' o 'pasa') o estar incluída en otros términos (como 'casado' o 'escasa'). Además, este sistema (y en general los sistemas STD) está especialmente pensado para el tipo de términos denominado como palabras Out-Of-Vocabulary por lo que es seguro que los términos que no son nombres propios cambiarán el rendimiento del sistema cuando estén incluídos en el diccionario de búsqueda.

Esta reestructuración de diccionarios de cara a analizar el impacto de esos términos 'especiales' se realiza para palabras largas (con las que se analiza la influencia de los términos en plural) y para palabras cortas (con las que se mira cómo influyen las palabras fácilmente confundibles o contenidas en otras). En ambos casos se usa sólo un diccionario original de 30 palabras, ya que se desea ver el efecto de la presencia o no de ese tipo de términos en el diccionario y no la diferencia entre el tamaño de los diccionarios.

4.2.5. Experimento 5: Utilización de una matriz de confusión

En el último experimento realizado se ha introducido el uso de una matriz de confusión del idioma español para realizar las búsquedas, teniendo en cuenta de esta forma unas determinadas probabilidades de inserción, borrado y sustitución que se pueden observar en el Anexo B, en el que se encuentra la matriz de confusión del idioma español proporcionada por Doroteo Torre Toledano [47]. El resto de parámetros fijados ha sido el uso de palabras cortas (ya que son las palabras con más riesgo de confusión), con un margen de score de -80 y un tamaño de diccionario de 30 palabras. Se fijan estos parámetros ya que se quiere ver el efecto de la matriz de confusión y sólamente eso.

4.3. Resultados obtenidos

4.3.1. Forma de expresar resultados

Para cada uno de los experimentos descritos en la sección anterior de este capítulo, se muestran en el siguiente apartado unas tablas que recogen las estadísticas más importantes fruto de la evaluación de las búsquedas comparadas en cada experimento. En dichas tablas se ve reflejada la totalidad de los siguientes valores:

- Número de referencias reales existentes en el audio (Ref).
- Número de coincidencias correctas encontradas por el sistema (Corr).
- Número de Falsas Alarmas (FA).
- Número de pérdidas (Miss).

Y por supuesto, para poder valorar el sistema dadas unas condiciones y compararlo con el resto de casos, para cada experimento se da también una tabla con:

- Número medio de Falsas Alarmas por segundo de audio (P(FA)).
- Probabilidad media de pérdida (P(Miss)).
- Actual Term-Weighted Value (ATWV).

En el caso del ATWV, no es fácil decidir si es un valor bueno o malo simplemente viéndolo, pero nos servirá para comparar casos probados dentro de un experimento y comparar también el sistema en el que se centra este proyecto con sistemas similares, sabiendo el valor ATWV de dichos sistemas.

Para poder ver gráficamente el comportamiento del sistema en los diferentes casos estudiados, se muestra también la curva DET obtenida de la evaluación de cada búsqueda que interviene en

los experimentos. En ella se puede ver la relación entre las Falsas Alarmas y las pérdidas que se han producido. Finalmente, se realiza una reflexión sobre los resultados observados en cada evaluación del experimento. Dichas reflexiones serán resumidas en el siguiente y último capítulo que engloba las conclusiones del proyecto.

Como complemento a los datos que se muestran en el siguiente apartado, es posible observar en el Anexo C las tablas con los resultados completos término a término de la evaluación de cada búsqueda realizada en los experimentos.

4.3.2. Tablas y gráficas de resultados

4.3.2.1. Experimento 1: Comparación de las distintas longitudes de palabras

En la Tabla 4.1 y Tabla 4.2 se muestran los resultados más importantes de este experimento y en la Figura 4.1 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas.

Palabras largas				
Estadísticas	Ref	Corr	FA	Miss
Total	224	44	88	188
Palabras mixtas				
Estadísticas	Ref	Corr	FA	Miss
Total	281	88	5507	193
Palabras cortas				
Estadísticas	Ref	Corr	FA	Miss
Total	284	119	14237	165

Tabla 4.1: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
Palabras largas	0.01425	0.804	**0.010**
Palabras mixtas	0.90013	0.687	**-9.405**
Palabras cortas	2.32821	0.581	**-24.32**

Tabla 4.2: Probabilidades de errores y ATWV

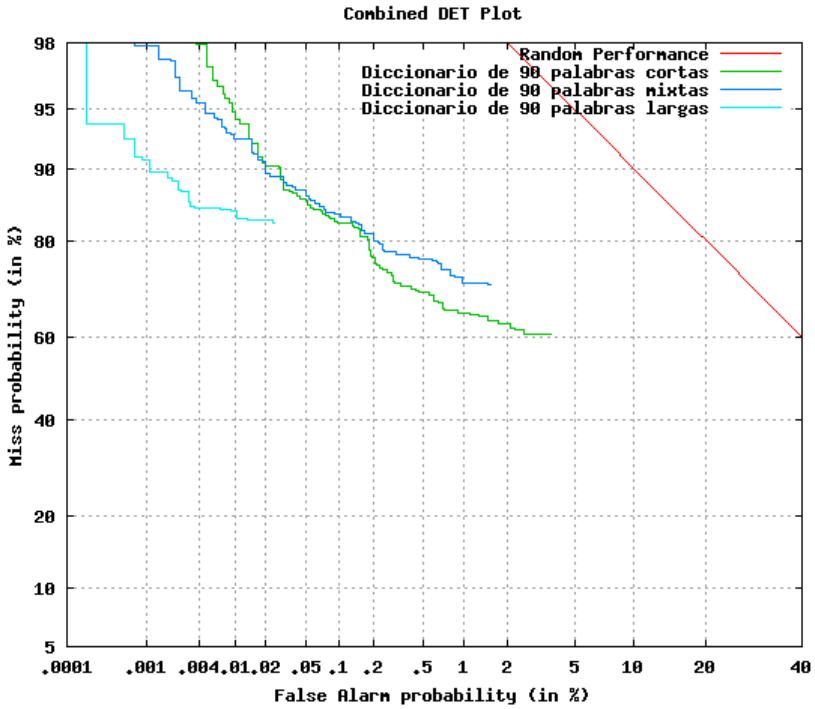

Figura 4.1: Curvas DET para palabras cortas, mixtas y largas

El mejor comportamiento del sistema se da para un diccionario de palabras largas seguido de las palabras mixtas y por último las palabras cortas. Esto se puede observar a través de distintos aspectos destacables obtenidos en los resultados del experimento:

- Una mejor proporción entre Falsas Alarmas y pérdidas en el caso de las palabras largas respecto a las cortas.
- Mejor relación de probabilidades de FA y de pérdida y por tanto mayor ATWV en el caso de palabras largas.
- Una curva DET más cercana al origen para el diccionario de palabras largas.

Este mejor comportamiento de las palabras largas respecto de las cortas se debe a la potencial capacidad de confusión de cada tipo de palabras. Mientras que una palabra larga como 'mediterráneo' se parece a pocas palabras y no es fácilmente confundible, una palabra corta como 'león' es similar a 'peón' o 'neón' y además está contenida en palabras como 'leonardo' o 'camaleón' y, por lo tanto, se puede confundir con ellas a la hora de buscarla en los lattices de fonemas reconocidos. De ahí se deriva principalmente la cantidad desproporcionada de FA que se producen para palabras cortas.

4.3.2.2. Experimento 2: Ajuste del umbral de score

En la Tabla 4.3 y Tabla 4.4 se muestran los resultados más importantes de este experimento y en la Figura 4.2 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas.

Sin umbral				
Estadísticas	Ref	Corr	FA	Miss
Total	224	45	140	179
Umbral = -250				
Estadísticas	Ref	Corr	FA	Miss
Total	224	44	88	180
Umbral = -225				
Estadísticas	Ref	Corr	FA	Miss
Total	224	44	74	180

Tabla 4.3: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
Sin umbral	0.02267	0.799	**-0.079**
Umbral = -250	0.01425	0.804	**0.010**
Umbral = -225	0.01198	0.804	**0.035**

Tabla 4.4: Probabilidades de errores y ATWV

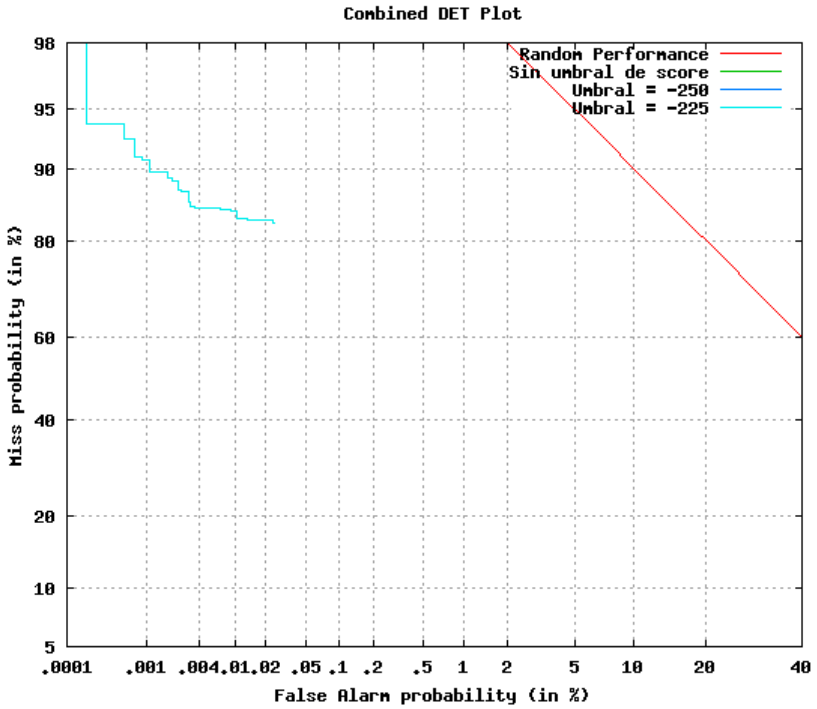

Figura 4.2: Curvas DET para distintos casos de umbral de score

Observando los datos anteriormente mostrados, destaca el hecho de la gran mejora en el valor ATWV según se va subiendo el margen impuesto para decidir si la coincidencia es válida o no en función del score. Los resultados mejoran al subir el umbral hasta un umbral óptimo (umbrales por encima o por debajo de él empeoran el ATWV) que para estas condiciones (diccionario de

90 palabras largas) ha resultado ser -225. Como contraste, se observa como la curva DET no varía al variar el umbral.

El motivo de mejora del ATWV y principal efecto de la optimización del citado umbral es la drástica reducción de las Falsas Alarmas, ya que ahora se descartan coincidencias que estaban obteniendo scores muy bajos (por debajo del umbral). Además, esto ocurre sin existir un aumento significativo de las pérdidas (tan sólo una más), ya que el score de las coincidencias correctas sí supera el umbral impuesto. El hecho de que la optimización de umbral no tenga efecto en la curva DET se debe a que las probabilidades de pérdida apenas cambian y las de Falsa Alarma, aunque cambian, son muy bajas como para que sea significativo.

Se ha realizado el mismo experimento con 90 palabras cortas obteniendo, no con tanta precisión sino llegando más bien a un compromiso entre FA y pérdidas, un valor del umbral 'óptimo' de -80. Este valor es más pequeño que para palabras largas porque hay muchas coincidencias correctas de palabras cortas con un score por debajo de -225 y que con ese umbral se 'tirarían'. Estos scores más bajos en las coincidencias correctas de palabras cortas son lógicos ya que puede haber más 'dudas' a la hora de puntuar la coincidencia, menos probabilidad de que sea correcta, dando lugar a esos scores.

4.3.2.3. Experimento 3: Comparación del tamaño de diccionario

En la Tabla 4.5 y Tabla 4.6 se muestran los resultados más importantes de este experimento y en la Figura 4.3 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas.

90 palabras				
Estadísticas	Ref	Corr	FA	Miss
Total	224	44	74	180
30 palabras				
Estadísticas	Ref	Corr	FA	Miss
Total	81	16	17	65

Tabla 4.5: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
90 palabras	0.01198	0.804	**0.035**
30 palabras	0.00269	0.802	**0.095**

Tabla 4.6: Probabilidades de errores y ATWV

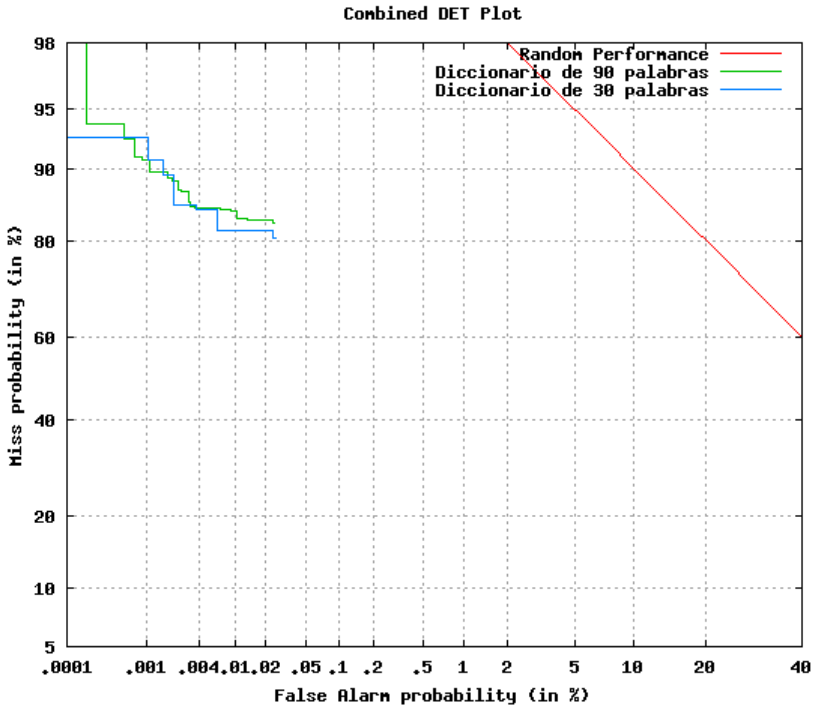

Figura 4.3: Curvas DET para diccionarios de 90 y 30 palabras

Las tablas y gráficas anteriores desvelan cómo tanto los valores de probabilidad de pérdida y falsas alarmas por segundo como el ATWV son algo mejores en el caso de un tamaño de diccionario de 30 palabras respecto al de 90. En las curvas DET apenas se aprecia dicha diferencia, simplemente la curva de 30 palabras es un poco mejor.

Estos resultados nos llevan a la conlusión de que el tamaño del diccionario de búsqueda no influye directamente en el rendimiento del sistema. Cualquier aparente mejora en los resultados (como en el caso de este experimento) será producto de la influencia de la naturaleza de los términos escogidos para elaborar los diccionarios (en este caso en concreto, al ser el diccionario de 30 palabras un extracto del de 90, se habrán eliminado, sin ningún criterio, algunas palabras que deteriorasen el rendimiento del sistema -el impacto de este tipo de palabras es analizado en el experimento siguiente-).

4.3.2.4. Experimento 4: Análisis de la influencia de los términos en plural y de los términos que pueden aparecer dentro de otros términos

En la Tabla 4.7 y Tabla 4.8 se muestran los resultados más importantes de este experimento y en la Figura 4.4 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas para ver el efecto de los téminos en plural en un diccionario de palabras largas (umbral -225).

Dicc. orig. 30 largas (con plurales)				
Estadísticas	Ref	Corr	FA	Miss
Total	81	16	17	65
Dicc. modif. 24 largas (sin plurales)				
Estadísticas	Ref	Corr	FA	Miss
Total	73	16	15	57

Tabla 4.7: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
Dicc. orig. 30 largas (con plurales)	0.00269	0.802	**0.095**
Dicc. modif. 24 largas (sin plurales)	0.00237	0.781	**0.132**

Tabla 4.8: Probabilidades de errores y ATWV

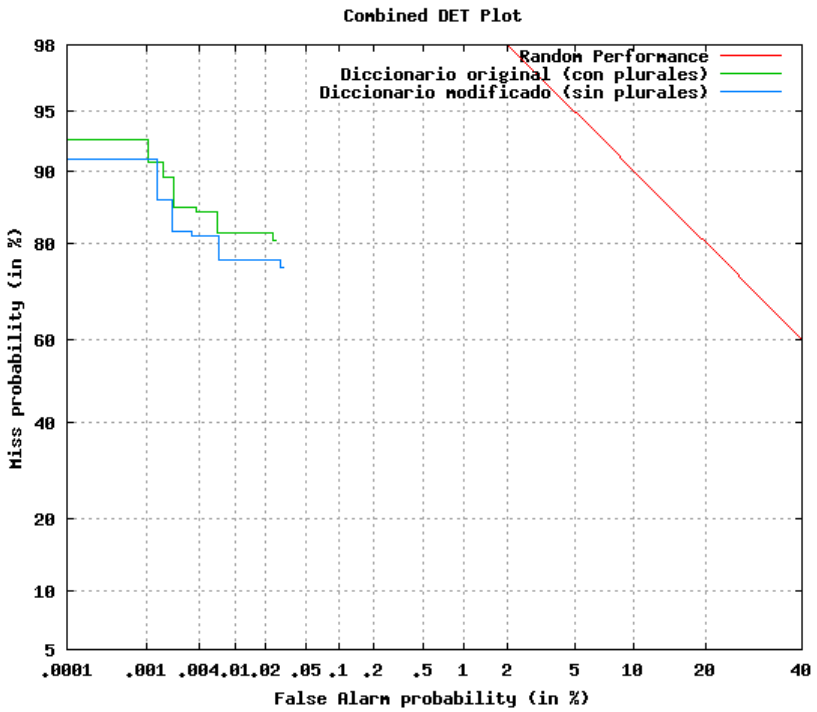

Figura 4.4: Curvas DET para ver el efecto de las palabras en plural

A la luz de estos resultados, se puede concluir que la presencia de palabras en plural en el diccionario de búsqueda tiene un impacto negativo en el rendimiento de las búsquedas, produce unos resultados pesimistas. Los resultados sin ellas en el diccionario son más optimistas, dando lugar a mejores valores de probabilidad de pérdida y ATWV así como una curva DET más cercana al origen. Esto es debido a que con ellas en el diccionario se producen más FA por la localización de esas palabras en singular como si fuera la palabra en plural. Este hecho no es en sí del todo malo y se podría optimizar el sistema para tener en cuenta que la palabra puede aparecer tanto en singular como en plural y contabilizar ambas como coincidencias correctas.

En la Tabla 4.9 y Tabla 4.10 se muestran los resultados más importantes de este experimento y en la Figura 4.5 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas para observar el efecto de términos confundibles en un diccionario de palabras cortas (umbral -80).

Dicc. orig. 30 cortas (con confundibles)				
Estadísticas	Ref	Corr	FA	Miss
Total	110	31	556	79
Dicc. modif. 27 cortas (sin confundibles)				
Estadísticas	Ref	Corr	FA	Miss
Total	99	27	183	72

Tabla 4.9: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
Dicc. orig. 30 cortas (con confundibles)	0.08841	0.718	**-2.648**
Dicc. modif. 27 cortas (sin confundibles)	0.02905	0.727	**-0.826**

Tabla 4.10: Probabilidades de errores y ATWV

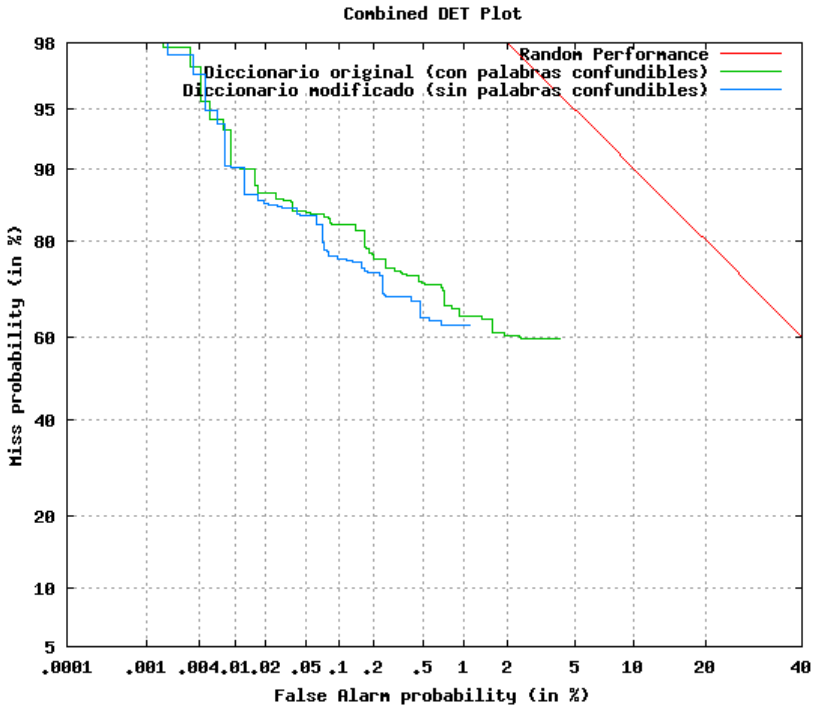

Figura 4.5: Curvas DET para ver el efecto de las palabras fácilmente confundibles

Analizando los anteriores resultados, se puede observar que la presencia de palabras fácilmente confundibles, contenidas en otras palabras o que no son nombres propios en el diccionario de búsqueda produce un peor rendimiento de las búsquedas. Sin ellas en el diccionario se dan mejores valores de probabilidad de pérdida y ATWV así como una curva DET más cercana al origen, especialmente para las probabilidades de FA más altas. Esto es debido a que con ellas en el diccionario se producen más FA por la supuesta localización de las palabras cuando en realidad es una que la contiene o que se le parece mucho.

Lo que ocurre en los dos subexperimentos realizados en este experimento número 4 es lógico ya que las palabras 'contaminadas' no acarrean ningún beneficio a las búsquedas. Queda comprobado por tanto que este sistema es ciertamente sensible a ese tipo de términos.

4.3.2.5. Experimento 5: Utilización de una matriz de confusión

En la Tabla 4.11 y Tabla 4.12 se muestran los resultados más importantes de este experimento y en la Figura 4.6 se ven las curvas DET originadas por la evaluación de las búsquedas realizadas.

Sin matriz de confusión				
Estadísticas	Ref	Corr	FA	Miss
Total	110	31	556	79
Con matriz de confusión				
Estadísticas	Ref	Corr	FA	Miss
Total	110	31	756	79

Tabla 4.11: Resumen de datos totales

Estadísticas	P(FA)	P(Miss)	ATWV
Sin matriz de confusión	0.08841	0.718	**-2.648**
Con matriz de confusión	0.12021	0.718	**-3.69**

Tabla 4.12: Probabilidades de errores y ATWV

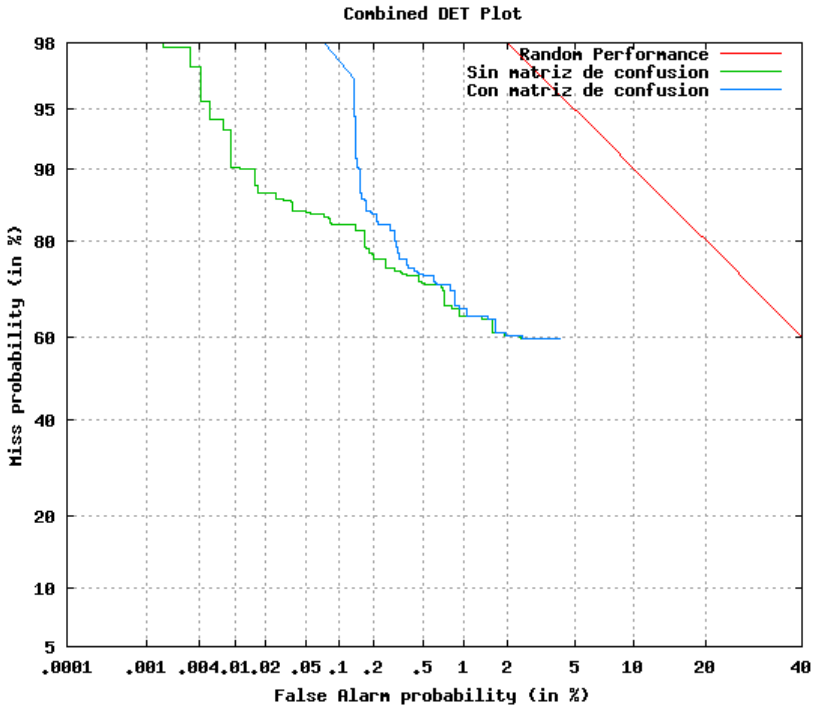

Figura 4.6: Curvas DET para ver la influencia de la matriz de confusión

Al introducir el uso de la matriz de confusión se ve que tanto el ATWV como la probabilidad de falsa alarma por segundo aumentan, mientras que en el nuevo escenario las pérdidas de términos se mantienen igual que sin usar la matriz de confusión. La curva DET también es peor pero es menos mala cuanto mayor es la probabilidad de FA y menor es la de pérdida.

La matriz de confusión debería venir bien para eliminar pérdidas, sin embargo esto no ocurre y lo que pasa es que dicha matriz actúa en demasía para algunas FA que sin la matriz se descartaban correctamente y que ahora el sistema califica como coincidencias (por culpa de las probabilidades de sustitución, inserción y borrado), produciendo así un aumento en las FA. De este experimento se saca en claro que el uso de una matriz de confusión para este sistema no es favorable, ya que no se consigue la disminución deseada de las pérdidas y por contra las falsas alarmas aumentan (recordemos que en este tipo de sistemas se les da preferencia a las pérdidas frente a las FA).

5

Conclusiones y trabajo futuro

5.1. Conclusiones

Tras un análisis de los resultados obtenidos en los experimentos realizados mostrados en el capítulo anterior de esta memoria, se ha llegado a una serie de conclusiones acerca del rendimiento del sistema de detección de términos orales desarrollado en este proyecto. Estas conclusiones versan principalmente sobre el comportamiento del sistema respecto a la variación de sus parámetros de entrada y se resumen a continuación:

- Si varían las longitudes de los términos a buscar, el mejor comportamiento del sistema se alcanza para un diccionario de palabras largas, mientras que el peor comportamiento se da para palabras cortas. Un diccionario de palabras largas y cortas mezcladas tiene un rendimiento intermedio.

- Existe un factor bastante decisivo de cara al aumento del ATWV del sistema (y por tanto de su rendimiento, dado que el ATWV nos permite comparar rendimientos). Ese factor es la inteligencia añadida al sistema por el hecho de incluir un decisor a la salida del sistema de búsqueda de términos que, en función del score de las coincidencias obtenidas, decide las que son válidas y las que no según un umbral.

- El número de palabras en el diccionario no influye directamente en el comportamiento del sistema. Más que el número de palabras, es la naturaleza de las mismas lo que puede llevar a variar el rendimiento del sistema.

- Los términos en plural, las palabras que no son nombres propios o acrónimos (es decir, que no son términos OOV) y las palabras fácilmente confundibles o contenidas en otras, influyen negativamente en el rendimiento del sistema reconocedor de palabras clave basado en STD.

- El uso de una matriz de confusión para las búsquedas realizadas con este sistema no supone un beneficio ya que las Falsas Alarmas aumentan considerablemente y las pérdidas se mantienen inalteradas, cosa que no resulta interesante ya que lo que se persigue en este tipo de sistemas es tener las menores pérdidas posibles.

Para comparar este sistema de detección de términos orales con otros sistemas de similar naturaleza se va a utilizar el ATWV ya que, a pesar de no definir completamente al sistema, nos permite una comparación bastante objetiva entre el rendimiento de distintos sistemas. El mejor ATWV obtenido en los experimentos realizados en el presente proyecto ha sido de 0.132 (para el caso de 30 palabras largas sin términos en plural). En los sistemas similares desarrollados en el Área de Tratamiento de Voz y Señales (ATVS) de la Universidad Autónoma de Madrid el máximo ATWV alcanzado ha sido del orden de 0.3, luego se puede decir que el rendimiento de este sistema no está lejos de los sistemas de detección de términos hasta ahora desarrollados.

En resumen, el sistema desarrollado en este proyecto tiene un rendimiento aceptable a la hora de detectar palabras OOV de una longitud grande. Gracias a estas características, este sistema podría ser un complemento ideal a un sistema de word spotting basado en un reconocedor de habla continua de gran vocabulario (LVCSR) al que ayudaría a encontrar, de una forma bastante eficiente, los términos fuera de vocabulario (OOV) que él es incapaz de localizar.

5.2. Trabajo futuro

Una vez comprendidos los resultados del análisis del sistema desarrollado, se deberían tener en cuenta las siguientes posibles mejoras para trabajos futuros que continúen desarrollando este sistema:

- El método de decisión usado para discriminar las coincidencias válidas de las erróneas mediante un umbral de score podría mejorarse teniendo en cuenta más factores que intervengan en la búsqueda y de ese modo discernir más eficientemente las veces que la búsqueda ha funcionado con éxito de las que no en el sistema decisor situado tras la búsqueda. Una posibilidad es el uso de una red neuronal diseñada para tal fin.

- El algoritmo de detección de términos podría tener un funcionamiento más optimizado de cara a palabras cortas, de forma que disminuyesen primordialmente las Falsas Alarmas en este tipo de términos.

- El sistema podría ser adaptado para poder usar otras sub-unidades de palabra distintas a los fonemas (grafemas, sílabas) como base de la búsqueda y poder así observar el comportamiento de STD en otras condiciones.

- Se podría realizar una pequeña funcionalidad mediante la cual el usuario sólo introduciría las palabras en el diccionario de búsqueda, sin escribir su transcripción fonética (ya que no la tiene por qué conocer). Sería función de una pequeña herramienta realizar la transcripción fonética y escribirla en el diccionario definitivo que se introducirá al sistema.

- Una posible mejora del sistema podría ser adaptar los diccionarios a las distintas formas de pronunciación de las palabras, de forma que cada palabra con más de una posible pronunciación apareciera tantas veces en el diccionario como pronunciaciones tenga, cada vez con una transcripción fonética distinta. Por ejemplo, el seseo característico del español de América da lugar a dos pronunciaciones de las palabras que contengan el sonido 'z', una representada por el fonema /T/ y otra por el fonema /s/. Esta mejora del sistema aparece por el desconocimiento a priori del acento del locutor. En este caso hablamos del idioma español pero sería aplicable a términos de cualquier idioma.

- Otra posible mejora sería adaptar el sistema para que considere coincidencias correctas de un término no sólo a ese término sino también a las palabras con género (masculino/femenino) y número (singular/plural) contrarios, en caso de existir.

La tecnología de la localización de términos orales tiene una proyección futura muy prometedora y una previsible rápida integración en el mercado. Esto es así debido a sus múltiples aplicaciones entre las que destacan la capacidad de encontrar palabras clave en el audio que se puede encontrar en Internet (el audio de los vídeos, por ejemplo) o la posibilidad de ayudar a las fuerzas y cuerpos de seguridad del estado a luchar contra el crimen mediante la búsqueda automática de términos potencialmente peligrosos (como 'bomba', 'atentado' o 'terrorista') en conversaciones telefónicas. En definitiva, son sistemas realmente útiles en la sociedad actual.

Glosario de términos

- **Alófono**: Es el término que se usa para definir cada uno de los sonidos de un idioma.
- **Alineador**: Sistema que consigue determinar el instante de inicio y de fin de cada fonema de un archivo de audio.
- **ATWV**: Actual Term-Weighted Value. Valor que caracteriza el rendimiento de un sistema de detección en el que existan falsas alarmas y pérdidas.
- **Curva DET**: Curva Detection Error Trade-off. Curva que representa la probabilidad de falsa alarma de un sistema de detección dada una probabilidad de pérdida, o viceversa.
- **Diccionario de búsqueda**: Lista de palabras y su correspondiente transcripción fonética que se desean buscar en un determinado audio.
- **DTW**: Dynamic Time Warping. Alineamiento temporal dinámico, técnica usada para el reconocimiento de voz que consiste en el alineamiento de un patrón y una secuencia de test.
- **FA**: False Alarm. Falsa Alarma, se produce cuando existe una detección de un término que realmente no aparece en el audio.
- **Fonema**: Unidad teórica básica creada para estudiar una lengua a nivel fónico-fonológico.
- **GNUPLOT**: Herramienta para la representación gráfica de funciones o vectores muy frecuentemente usada en el sistema operativo Linux.
- **Grafema**: Unidad lingüística mínima de la lengua escrita. Son las letras, números y símbolos lingüísticos.
- **Grafón**: Fragmento de una palabra que se entrena a partir de una secuencia de fonemas y grafemas de la palabra.
- **HMM**: Hidden Markov Model. Modelo oculto de Markov, modelo estadístico en el que se asume que el sistema a modelar es un proceso de Markov de parámetros desconocidos. Nos ayudamos de ellos para realizar reconocimiento de voz.
- **HTK**: Hidden Markov Model ToolKit. Kit de herramientas para la creación y el tratamiento de HMMs.
- **Keyword**: Palabra clave, término buscado por el usuario.
- **Lattice**: Entramado de fonemas reconocidos y probabilidades de saltar de uno a otro que forman un grafo acíclico dirigido que representa las secuencias de fonemas más probables en el audio.
- **LVCSR**: Large Vocabulary Continuous Speech Recognition. Reconocimiento de habla continua de gran vocabulario, tipo de sistema de word spotting basado en un diccionario completo de la lengua en concreto.

- **MA2VICMR**: Mejorando el Acceso, el Análisis y la Visibilidad de la Información y los Contenidos Multilingüe y Multimedia en Red.
- **Matriz de confusión**: Matriz en la que se reflejan las probabilidades de inserción, borrado y sustitución de los fonemas dentro de un lattice.
- **Miss**: Del inglés pérdida, se produce cuando no se detecta un término que existe en el audio.
- **MLF**: Master Label File. Archivo de etiquetas usado en HTK.
- **NIST**: National Institute of Standards and Technology. Organismo federal no regulador, perteneciente a la Cámara de Comercio de los Estados Unidos, que desarrolla y promueve medidas, estándares y tecnología para aumentar la productividad, facilitar el comercio y mejorar la calidad de vida.
- **OOV**: Out-Of-Vocabulary. Palabras fuera de vocabulario, usualmente nombres propios o acrónimos.
- **PERL**: Practical Extraction and Report Language. Lenguaje de programación que toma características del lenguaje C y es interpretado por un intérprete de comandos.
- **RTTM**: Rich Transcription Time Mark. Formato de archivo que contiene el alineamiento de un audio palabra por palabra.
- **Score**: Puntuación que define la calidad de una detección.
- **ShellScripting**: Lenguaje de programación basado en comandos del intérprete de comandos de Linux.
- **STD**: Spoken Term Detection. Detección de términos orales, técnica basada en lattices de sub-unidades de palabra usada para la localización de palabras clave en audio.
- **Transcriptor**: Sistema que logra separar cada palabra de un audio en sub-unidades (fonemas, sílabas).
- **Umbral**: Valor constante que se utiliza para discernir cuando un resultado cumple o no unos requisitos mínimos en un sistema de detección.
- **VQ**: Vectorial Quantification. Cuantificación vectorial, técnica de reconocimiento de audio en la que se calculan unos centroides mediante el entrenamiento del sistema y se mide la distancia de los vectores de test a los centroides eligiendo la menor de ellas.
- **Word Spotting**: Es la ciencia que estudia la localización de palabras clave en audio.

Bibliografía

[1] http://arantxa.ii.uam.es/ jms/pfcsteleco/lecturas/20110926JoseAntonioMorejonSaravia.pdf

[2] L.E. Baum: An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of Markov Processes, Inequalities, vol. 3, pp. 1-8, 1972.

[3] U.K. Baker: The DRAGON System - An Overview, IEEE Trans. ASSP, vol. 23, pp. 24-29, Febrero 1975.

[4] R. Bakis: Continuous Speech Recognition via Centisecond Acoustic States, 91st Meeting of the Acoustical Society of America, Abril 1976.

[5] L.R. Rabiner: A tutorial on hidden markov models and selected applications in speech recognition. IEEE, 77, 2:257-286, 1989.

[6] Andrew J. Viterbi: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory, 13, 2:260-269, 1967.

[7] L. E. Baum, T. Petrie, G. Soules, N. Weiss: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann. Math. Statist., 41:164-171, 1970.

[8] Javier Tejedor Noguerales, Doroteo T. Toledano, José Colás Pasamontes: Estado del arte en Wordspotting aplicado a los sistemas de extracción de información en contenidos de voz. I Congreso Español de Recuperación de Información (CERI). Madrid, España. Junio 2010.

[9] Logan, B., Moreno, P., Van Thong, J.M., Whittaker, E.: An experimental study of an audio indexing system for the web. ICASSP, 2, 676-679, 2000.

[10] Watson, D.: Death sentence, The decay of public language. Knopf. Sydney, 2003.

[11] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon : Spoken Language Processing. Prentice Hall, 2001.

[12] Rose, R.C., Paul, D.B.: A hidden markov model based keyword recognition system. ICASSP, 129-132, 1990.

[13] Manos, A.S., Zue, V.W.: A segment-based wordspotter using phonetic filler models. ICASSP, 2, 899-902, 1997.

[14] Cuayahuitl, H., Serridge, B.: Out-of-vocabulary word modelling and rejection for spanish keyword spotting systems. MICAI, 155-165, 2002.

[15] Xin, L., Wang, B.: Utterance verification for spontaneous mandarin speech keyword spotting. ICH, 3, 397-401, 2001.

[16] Tejedor, J., King, S., Frankel, J., Wang, D., Colás, J.: A novel two-level architecture plus confidence measures for a keyword spotting system. V Jornadas en Tecnología del Habla, 247-250, 2008.

[17] Szoke, I., Schwarz, P., Matejka, P., Burget, L., Karafiat, M., Fapso, M., Cernocky, J.: Comparison of keyword spotting approaches for informal continuous speech. ICSLP, 633-636, 2005.

[18] Tejedor, J., García R., Fernández, M., López-Colino, F., Perdrix, F., Macías J.A. Gil, R.M., Oliva, M., Moya, D., Colás, J., Castells, P.: Ontology-based retrieval of human speech. Proc. of DEXA, 485-489, 2007.

[19] Ou, J., Chen, C., Li, Z.: Hybrid neural-network/HMM approach for out-of- vocabulary words rejection in mandarin place name recognition. ICONIP, 2001.

[20] Ben Ayed, Y., Fohr, D., Haton, J.P., Chollet, G.: Keyword spotting using support vector machines. TSD, 285-292, 2002.

[21] Ben Ayed, Y., Fohr, D., Haton, J.P., Chollet, G.: Confidence measures for keyword spotting using support vector machines. ICASSP, 1, 588-591, 2003.

[22] NIST.: The spoken term detection (STD) 2006 evaluation plan. http://www.nist.gov/speech/tests/std.

[23] Amir, A., Efrat, A., Srinivassan, S.: Advances in phonetic word spotting. CIKM, 580-582, 2001.

[24] Thambiratnam, K., Sridharan, S.: Rapid yet accurate speech indexing using dynamic match lattice spotting. IEEE Trans. on Audio and Speech Processing, 15(1), 346-357, 2007.

[25] Tejedor, J., Wang, D., Frankel, J., King. S, Colás, J.: A comparison of grapheme and phoneme-based units for spanish spoken term detection. Speech Communication, 50(11-12), 980-991, 2008.

[26] Wang, D., Frankel, J., Tejedor, J., Colás, J.: Comparison of phone and grapheme-based spoken term detection. ICASSP, 4969-4972, 2008.

[27] Akbacak, M., Vergyri, D., Stolcke, A.: Open-vocabulary spoken term detection using graphone-based hybrid recognition systems. ICASSP, 5240-5243, 2008.

[28] Larson, M., Eickeler, S., Kohler, J.: Supporting radio archive workflows with vocabulary independent spoken keyword search. SSCS-SIGIR, 2007.

[29] Yu, P., Seide, F.: A hybrid word/phoneme-based approach for improved vocabulary-independent search in spontaneous speech. ICSLP, 635-643, 2004.

[30] Iwata, K., Shinoda, K., Furui, S.: Robust spoken term detection using combination of phone-based and word-based recognition. Interspeech, 2195-2198, 2008.

[31] Szoke, I., Fapso, M., Burget, L., Cernocky, J.: Hybrid word-subword decoding for spoken term detection. SSCS-SIGIR, 2008.

[32] Vergyri, D., Shafran, I., Stolcke, A., Gadde, R.R., Akbacak, M., Roark, B., Wang, W.: The SRI/OGI 2006 spoken term detection system. Interspeech, 2393-2396, 2007.

[33] Miller, D.H.R., Kleber, M., Lin Kao, C., Kimball, O., Colthurst, T., Lowe, S.A., Schwartz, R.M., Gish, H.: Rapid and accurate spoken term detection. Interspeech, 314-317, 2007.

[34] Javier Tejedor, Dong Wang, Joe Frankel, Simon King, José Colás: A comparison of grapheme and phoneme-based units for Spanish spoken term detection. Speech Communication, 50, 980Ű991, 2008.

[35] Murat Akbacak, Dimitra Vergyri, Andreas Stolcke: Open-vocabulary Spoken Term Detection using graphone-based hybrid recognition systems. Speech Technology and Research Laboratory SRI International, IEEE, ICASSP, USA, 2008.

[36] http://www.mavir.net/

[37] http://www.madrimasd.org/informacionidi/programas/ficha.aspx?idproyecto=2027

[38] http://mavir2006.mavir.net/

[39] http://www.csic.es/

[40] http://www.uam.es/

[41] http://www.uc3m.es/

[42] http://www.uem.es/

[43] http://www.uned.es/

[44] http://www.upm.es/

[45] http://www.urjc.es/

[46] http://www.itl.nist.gov/iad/mig//tests/std/datainfo.html

[47] Doroteo Torre Toledano: Estimación a priori del grado de confusión entre palabras para reconocedores fonéticos de vocabulario flexible: métodos y aplicaciones. Proyecto fin de Carrera. Departamento de Señales, Sistemas y Radiocomunicaciones Universidad Politécnica de Madrid. Director: Luis Hernández Gómez. Madrid, España. 1997.

A

Diccionarios de búsqueda

Por motivos de comodidad de representación, sólo se muestran los diccionarios de 30 palabras usados ya que los de 90 palabras son excesivamente largos. Estos últimos tienen la misma estructura que los que se muestran, simplemente aparecen más palabras.

A1. Diccionario de búsqueda de palabras largas

```
universidad.    u n i b e r s i d a d
renacentistas.  R e n a T e n t i s t a s
arquitectura.   a r k i t e k t u r a
restaurantes.   R e s t a u r a n t e s
ayuntamiento.   a y u n t a m i e n t o
barcelona.      b a r T e l o n a
modernismo.     m o d e r n i s m o
plateresca.     p l a t e r e s k a
compostela.     k o m p o s t e l a
hospitales.     o s p i t a l e s
guadalajara.    g u a d a l a j a r a
cristianos.     k r i s t i a n o s
delirante.      d e l i r a n t e
guadarrama.     g u a d a R a m a
extremadura.    e x t r e m a d u r a
cantábrico.     k a n t a b r i k o
mudéjares.      m u d e j a r e s
mediterráneo.   m e d i t e R a n e o
tenerife.       t e n e r i f e
cervantino.     T e r b a n t i n o
pinacotecas.    p i n a k o t e k a s
catalunya.      k a t a l u N a
albarracín.     a l b a R a T i n
capiteles.      k a p i t e l e s
salamanca.      s a l a m a n k a
jerusalén.      j e r u s a l e n
acueducto.      a k u e d u k t o
isabelino.      i s a b e l i n o
agricultura.    a g r i k u l t u r a
velázquez.      b e l a T k e T
```

A2. Diccionario de búsqueda de palabras mixtas

```
universidad.    u n i b e r s i d a d
barcelona.      b a r T e l o n a
compostela.     k o m p o s t e l a
guadalajara.    g u a d a l a j a r a
guadarrama.     g u a d a R a m a
extremadura.    e x t r e m a d u r a
cantábrico.     k a n t a b r i k o
mediterráneo.   m e d i t e R a n e o
tenerife.       t e n e r i f e
cervantino.     T e r b a n t i n o
albarracín.     a l b a R a T i n
salamanca.      s a l a m a n k a
jerusalén.      x e r u s a l e n
acueducto.      a k u e d u k t o
isabelino.      i s a b e l i n o
madrid. .       m a d r i d
córdoba. .      k o r d o b a
picasso. .      p i k a s o
gaudí.  .       g a u d i
cáceres. .      k a T e r e s
león.   .       l e o n
teruel. .       t e r u e l
cuenca. .       k u e n k a
mérida. .       m e r i d a
astorga. .      a s t o r g a
antoni. .       a n t o n i
ibiza.  .       i b i T a
platón. .       p l a t o n
quijote. .      k i j o t e
goya.   .       g o y a
```

A3. Diccionario de búsqueda de palabras cortas

```
madrid. .       m a d r i d
córdoba. .      k o r d o b a
españa. .       e s p a N a
ávila.  .       a b i l a
baeza.  .       b a e T a
picasso. .      p i k a s o
gaudí.  .       g a u d i
cáceres. .      k a T e r e s
santiago.       s a n t i a g o
león.   .       l e o n
teruel. .       t e r u e l
cuenca. .       k u e n k a
sofía.  .       s o f i a
civil.  .       T i b i l
mérida. .       m e r i d a
astorga. .      a s t o r g a
antoni. .       a n t o n i
ibiza.  .       i b i T a
platón. .       p l a t o n
quijote. .      k i j o t e
goya.   .       g o y a
joan.   .       y o a n
ainsa.  .       a i n s a
somport. .      s o m p o r t
colón.  .       k o l o n
martín. .       m a r t i n
segovia. .      s e g o b i a
greco.  .       g r e k o
úbeda.  .       u b e d a
valencia.       b a l e n T i a
```

A4. Diccionario de búsqueda de palabras largas eliminando los términos en plural

```
universidad.    u n i b e r s i d a d
arquitectura.   a r k i t e k t u r a
ayuntamiento.   a y u n t a m i e n t o
barcelona.      b a r T e l o n a
modernismo.     m o d e r n i s m o
plateresca.     p l a t e r e s k a
compostela.     k o m p o s t e l a
guadalajara.    g u a d a l a j a r a
cristianos.     k r i s t i a n o s
delirante.      d e l i r a n t e
guadarrama.     g u a d a R a m a
extremadura.    e x t r e m a d u r a
cantábrico.     k a n t a b r i k o
mediterráneo.   m e d i t e R a n e o
tenerife.       t e n e r i f e
cervantino.     T e r b a n t i n o
catalunya.      k a t a l u N a
albarracín.     a l b a R a T i n
salamanca.      s a l a m a n k a
jerusalén.      j e r u s a l e n
acueducto.      a k u e d u k t o
isabelino.      i s a b e l i n o
agricultura.    a g r i k u l t u r a
velázquez.      b e l a T k e T
```

A5. Diccionario de búsqueda de palabras cortas eliminando los términos fácilmente confundibles

```
madrid. .       m a d r i d
córdoba. .      k o r d o b a
españa. .       e s p a N a
ávila. .        a b i l a
baeza. .        b a e T a
picasso. .      p i k a s o
gaudí. .        g a u d i
cáceres. .      k a T e r e s
santiago.       s a n t i a g o
teruel. .       t e r u e l
cuenca. .       k u e n k a
sofía. .        s o f i a
civil. .        T i b i l
astorga. .      a s t o r g a
antoni. .       a n t o n i
ibiza. .        i b i T a
platón. .       p l a t o n
quijote. .      k i j o t e
goya. .         g o y a
joan. .         y o a n
ainsa. .        a i n s a
somport. .      s o m p o r t
colón. .        k o l o n
martín. .       m a r t i n
segovia. .      s e g o b i a
greco. .        g r e k o
valencia.       b a l e n T i a
```

B

Matriz de confusión

%	/a/	/b/	/T/	/C/	/d/	/e/	/f/	/g/	/i/	/j/	/k/	/l/	/L/	/m/	/n/	/ll/	/o/	/p/	/r/	/R/	/s/	/t/	/u/	/x/	Del
/a/	79,75	0,17	0,34	0,05	0,3	0,94	0,01	0,17	0,01	0,91	0,47	0,42	0,02	0,01	0,25	0,39	4,68	0,17	1,64	2,05	0,3	0,13	0,01	0,08	6,73
/b/	0,08	36,07	0,2	0,05	12,43	0,07	0,06	3,77	0,06	0,11	0,14	2,45	0,07	3,37	0,21	0,05	0,11	4,89	1,84	2,85	0,1	1,63	1,32	0,05	28,03
/T/	1,48	0,07	50,36	1,56	0,15	0,05	7,06	0,3	0,05	2,01	0,11	0,15	0,05	0,04	0,15	0,04	1,56	0,1	1,56	2,45	5,79	1,63	0,04	2,45	20,8
/C/	0,52	0,52	6,19	71,14	0,52	0,52	1,03	1,03	0,52	1,03	1,03	0,52	0,52	0,52	0,52	0,52	0,52	0,52	1,03	1,03	3,09	4,12	0,52	0,52	2,06
/d/	0,09	5,18	3,07	0,36	32,74	0,08	0,07	1,3	0,08	0,13	0,17	3,17	0,08	1,65	1,88	0,12	0,13	2,24	5,77	2,71	0,12	4,83	0,07	0,06	33,91
/e/	1,77	0,05	0,11	0,13	0,05	70,2	0,03	0,06	5,44	0,06	0,19	1,65	0,04	0,02	1,9	0,35	1,07	0,07	1,84	0,04	0,05	0,03	0,03	0,02	14,81
/f/	0,56	0,56	27,05	0,28	1,12	0,56	45,08	0,56	0,28	5,07	0,28	0,28	0,28	0,28	0,28	0,28	0,28	1,12	1,12	4,51	3,94	3,94	0,28	1,69	0,28
/g/	0,1	5,38	0,25	0,26	5	0,13	0,39	32,31	0,52	1,92	4,88	0,26	1,41	2,7	2,05	0,06	0,13	0,16	2,05	2,82	0,12	1,28	1,15	0,06	34,62
/i/	0,02	0,02	0,02	0,25	0,02	7,52	0,02	0,54	72,54	0,02	0,02	1,98	8,3	0,02	2,34	3,33	0,02	0,02	0,29	0,02	0,16	0,02	0,16	0,04	2,3
/j/	0,34	0,34	5,19	0,84	0,08	0,17	4,36	2,84	0,34	74,68	2,68	0,34	0,08	0,34	0,08	0,08	0,34	0,08	0,51	0,51	2,01	0,34	0,34	0,08	3,01
/k/	0,05	0,2	0,2	0,91	0,05	0,05	0,3	0,3	0,05	1,7	69,22	0,3	0,05	0,05	0,09	0,05	0,2	4,01	0,05	0,2	0,2	8,61	0,05	0,3	12,82
/l/	0,06	2,96	0,15	0,03	4,32	1,18	0,04	0,6	1,51	0,08	0,11	48,84	2,74	3,67	6,2	0,16	0,08	0,33	3,18	1,35	0,07	0,11	1,67	0,03	20,54
/L/	0,26	0,26	0,26	2,05	0,26	3,07	0,26	0,26	3,07	0,26	0,26	5,12	61,68	0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26	0,26	19,95
/m/	0,05	8,6	0,05	0,05	2,7	0,05	0,05	2,8	0,4	1	0,05	4,09	0,05	61,54	8,79	0,2	0,3	1,39	0,4	0,8	0,3	0,6	1,8	0,05	3,89
/n/	0,05	0,23	0,14	0,14	2,29	0,05	0,04	1,87	1,47	0,08	0,1	5,57	0,05	5,71	56,12	2,42	0,08	0,09	3,2	0,14	0,07	0,05	0,04	0,02	19,97
/ll/	0,45	0,45	0,45	0,45	0,45	1,8	0,45	0,45	0,9	0,9	0,45	0,9	3,6	0,45	0,9	82,89	0,45	0,45	0,45	0,45	0,45	0,45	0,45	0,45	0,45
/o/	9,27	0,28	0,16	0,12	0,52	0,2	0,02	0,84	0,02	0,52	0,02	0,28	0,02	0,16	1,92	0,12	70,34	0,02	0,4	1,72	0,02	0,02	8,63	0,04	4,35
/p/	0,17	2,47	0,12	0,66	0,17	0,08	0,17	0,33	0,08	0,08	2,79	0,17	0,08	0,08	0,12	0,08	0,08	64,36	0,08	0,08	0,08	10,86	0,08	0,08	16,62
/r/	1,13	0,12	2,85	0,31	6,09	3,5	0,07	0,14	0,08	0,14	0,18	3,93	1,47	0,06	3,06	0,02	0,14	0,17	30,15	4,06	4,23	2,03	0,07	0,13	35,89
/R/	0,13	2,62	1,57	0,53	4,45	0,26	0,53	4,19	0,13	0,79	6,02	1,83	0,26	3,66	0,26	0,13	0,26	2,36	3,66	60,49	0,79	2,88	0,26	0,13	1,83
/s/	0,04	0,05	17,2	4,02	0,05	0,04	5,77	0,29	0,03	1,28	0,12	0,18	0,04	0,03	0,11	0,03	0,06	0,12	1,98	0,35	51,49	0,03	0,03	1,86	14,81
/t/	0,05	0,31	0,41	2,46	1,85	0,05	0,41	0,05	0,05	0,31	4,42	0,05	0,05	0,05	0,31	0,05	0,05	9,87	0,21	0,05	0,05	73,84	0,05	0,05	4,94
/u/	0,05	2,41	0,1	0,05	0,05	0,05	0,05	1,26	0,05	0,06	0,07	1,99	0,05	1,26	0,11	0,21	6,8	0,07	0,06	1,26	0,05	0,05	69,17	0,05	14,65
/x/	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	1,74	50	10
Inser	1,75	14,08	16,6	5,18	11,69	3,05	17,59	21,52	3,5	16,98	16,3	6,13	15,87	5,51	12,38	4,24	5,53	19,55	10,6	12,17	9,53	6,76	6,56	13,68	-

C

Tablas de resultados completos

Por motivos de comodidad de representación, sólo se muestran los resultados para los diccionarios de 30 palabras, ya que para los de 90 palabras las tablas son excesivamente grandes. Estas últimas tienen la misma estructura que las que se muestran, simplemente aparecen más palabras.

C1. Palabras largas (umbral score = -250)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	universidad	9	3	0	6	0.333	0.000	0.667
TEST-02	renacentistas	2	0	2	2	-0.1	0.00031	1.0
TEST-03	arquitectura	15	1	0	14	0.067	0.000	0.933
TEST-04	restaurantes	1	0	0	1	0.000	0.000	1.0
TEST-05	ayuntamiento	4	1	0	3	0.250	0.000	0.750
TEST-06	barcelona	4	2	3	2	0.425	0.00047	0.500
TEST-07	modernismo	2	2	1	0	0.950	0.00016	0.0
TEST-08	plateresca	3	0	0	3	0.000	0.000	1.000
TEST-09	compostela	7	0	1	7	-0.014	0.00016	1.000
TEST-10	hospitales	1	0	0	1	0.000	0.000	1.000
TEST-11	guadalajara	2	0	0	2	0.000	0.000	1.000
TEST-12	cristianos	5	1	0	4	0.200	0.000	0.800
TEST-13	delirante	1	0	5	1	-0.500	0.00078	1.0
TEST-14	guadarrama	2	1	4	1	0.300	0.00063	0.500
TEST-15	extremadura	2	0	0	2	0.000	0.000	1.0
TEST-16	cantábrico	1	0	1	1	-0.100	0.00016	1.0
TEST-17	mudéjares	1	0	0	1	0.000	0.000	1.000
TEST-18	mediterráneo	1	1	1	0	0.9	0.00016	0.0
TEST-19	tenerife	1	0	1	1	-0.100	0.00016	1.000
TEST-20	cervantino	1	0	0	1	0.000	0.000	1.000
TEST-21	pinacotecas	1	0	0	1	0.000	0.000	1.000
TEST-22	catalunya	1	0	0	1	0.000	0.000	1.000
TEST-23	albarracín	1	0	0	1	0.000	0.000	1.000
TEST-24	capiteles	2	0	0	2	0.000	0.000	1.000
TEST-25	salamanca	3	2	0	1	0.667	0.000	0.333
TEST-26	jerusalén	2	0	0	2	0.000	0.000	1.000
TEST-27	acueducto	1	0	0	1	0.000	0.000	1.000
TEST-28	isabelino	2	2	1	0	0.950	0.00016	0.000
TEST-29	agricultura	1	0	1	1	-0.100	0.00016	1.0
TEST-30	velázquez	2	0	2	2	-0.100	0.00031	1.0
Totales / Occ-weighted Value		81	16	23	65	0.169	0.00364	0.802
Valores medios		2	0	0	2	0.134	0.00012	0.816
ATWV		**0.064**						

C2. Palabras mixtas (umbral score = -250)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	universidad	9	3	0	6	0.333	0.00000	0.667
TEST-02	barcelona	4	2	3	2	0.425	0.00047	0.500
TEST-03	compostela	7	0	1	7	-0.014	0.00016	1.0
TEST-04	guadalajara	2	0	0	2	0.000	0.00000	1.0
TEST-05	guadarrama	2	1	4	1	0.300	0.00063	0.500
TEST-06	extremadura	2	0	0	2	0.000	0.00000	1.0
TEST-07	cantábrico	1	0	1	1	-0.100	0.00016	1.0
TEST-08	mediterráneo	1	1	1	0	0.900	0.00016	0.000
TEST-09	tenerife	1	0	1	1	-0.100	0.00016	1.0
TEST-10	cervantino	1	0	0	1	0.000	0.00000	1.0
TEST-11	albarracín	1	0	0	1	0.000	0.00000	1.0
TEST-12	salamanca	3	2	0	1	0.667	0.00000	0.333
TEST-13	jerusalén	2	0	0	2	0.000	0.00000	1.0
TEST-14	acueducto	1	0	0	1	0.000	0.00000	1.0
TEST-15	isabelino	2	2	1	0	0.950	0.00016	0.000
TEST-16	madrid	5	2	26	3	-0.120	0.00407	0.600
TEST-17	córdoba	2	2	13	0	0.350	0.00203	0.000
TEST-18	picasso	3	0	0	3	0.000	0.00000	1.0
TEST-19	gaudí	15	11	88	4	0.147	0.01378	0.267
TEST-20	cáceres	7	3	26	4	0.057	0.00407	0.571
TEST-21	león	3	3	3141	0	-103.700	0.49109	0.000
TEST-22	teruel	1	0	19	1	-1.900	0.00297	1.0
TEST-23	cuenca	4	1	25	3	-0.375	0.00391	0.750
TEST-24	mérida	2	1	161	1	-7.550	0.02517	0.500
TEST-25	astorga	1	0	8	1	-0.800	0.00125	1.0
TEST-26	antoni	2	1	66	1	-2.800	0.01032	0.500
TEST-27	ibiza	4	1	15	3	-0.125	0.00235	0.750
TEST-28	platón	1	0	2	1	-0.200	0.00031	1.0
TEST-29	quijote	2	0	0	2	0.000	0.00000	1.0
TEST-30	goya	2	0	10	2	-0.500	0.00156	1.0
Totales / Occ-weighted Value		93	36	3612	57	-3.497	0.57279	0.613
Valores medios		3	1	120	1	-3.805	0.01883	0.698
ATWV		**-18.521**						

C3. Palabras cortas (umbral score = -250)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	madrid	5	2	26	3	-0.120	0.00407	0.600
TEST-02	córdoba	2	2	13	0	0.350	0.00203	0.000
TEST-03	españa	8	0	0	8	0.000	0.00000	1.000
TEST-04	ávila	4	4	226	0	-4.650	0.03534	0.000
TEST-05	baeza	6	4	44	2	-0.067	0.00688	0.333
TEST-06	picasso	3	0	0	3	0.000	0.00000	1.000
TEST-07	gaudí	15	11	88	4	0.147	0.01378	0.267
TEST-08	cáceres	7	3	26	4	0.057	0.00407	0.571
TEST-09	santiago	13	6	0	7	0.462	0.00000	0.538
TEST-10	león	3	3	3141	0	-103.7	0.49109	0.000
TEST-11	teruel	1	0	19	1	-1.900	0.00297	1.000
TEST-12	cuenca	4	1	25	3	-0.375	0.00391	0.750
TEST-13	sofía	2	0	9	2	-0.450	0.00141	1.000
TEST-14	civil	3	3	62	0	-1.067	0.00969	0.000
TEST-15	mérida	2	1	161	1	-7.550	0.02517	0.500
TEST-16	astorga	1	0	8	1	-0.800	0.00125	1.000
TEST-17	antoni	2	1	66	1	-2.800	0.01032	0.500
TEST-18	ibiza	4	1	15	3	-0.125	0.00235	0.750
TEST-19	platón	1	0	2	1	-0.200	0.00031	1.000
TEST-20	quijote	2	0	0	2	0.000	0.00000	1.000
TEST-21	goya	2	0	10	2	-0.500	0.00156	1.000
TEST-22	joan	1	1	192	0	-18.200	0.03001	0.000
TEST-23	ainsa	1	1	203	0	-19.300	0.03173	0.000
TEST-24	somport	1	0	1	1	-0.100	0.00016	1.000
TEST-25	colón	2	0	112	2	-5.600	0.01751	1.000
TEST-26	martín	2	0	98	2	-4.900	0.01532	1.000
TEST-27	segovia	3	1	4	2	0.200	0.00063	0.667
TEST-28	greco	3	0	39	3	-1.300	0.00610	1.000
TEST-29	úbeda	6	4	558	2	-8.633	0.08728	0.333
TEST-30	valencia	1	1	17	0	-0.700	0.00266	0.000
Totales / Occ-weighted Value		110	50	5165	60	-4.241	0.82128	0.545
Valores medios		3	1	172	2	-6.061	0.02692	0.594
ATWV		**-26.510**						

C4. Palabras largas sin umbral score

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	universidad	9	3	0	6	0.333	0.000	0.667
TEST-02	renacentistas	2	0	2	2	-0.100	0.00031	1.0
TEST-03	arquitectura	15	1	0	14	0.067	0.000	0.933
TEST-04	restaurantes	1	0	0	1	0.000	0.000	1.0
TEST-05	ayuntamiento	4	1	0	3	0.250	0.000	0.750
TEST-06	barcelona	4	2	3	2	0.425	0.00047	0.500
TEST-07	modernismo	2	2	2	0	0.900	0.00031	0.000
TEST-08	plateresca	3	0	0	3	0.000	0.000	1.0
TEST-09	compostela	7	0	1	7	-0.014	0.00016	1.0
TEST-10	hospitales	1	0	0	1	0.000	0.000	1.0
TEST-11	guadalajara	2	0	0	2	0.000	0.000	1.0
TEST-12	cristianos	5	1	0	4	0.200	0.000	0.800
TEST-13	delirante	1	0	9	1	-0.900	0.00141	1.0
TEST-14	guadarrama	2	1	20	1	-0.500	0.00313	0.5
TEST-15	extremadura	2	0	0	2	0.000	0.000	1.0
TEST-16	cantábrico	1	0	1	1	-0.100	0.00016	1.0
TEST-17	mudéjares	1	0	0	1	0.000	0.000	1.0
TEST-18	mediterráneo	1	1	1	0	0.900	0.00016	0.000
TEST-19	tenerife	1	0	2	1	-0.200	0.00031	1.0
TEST-20	cervantino	1	0	0	1	0.000	0.000	1.0
TEST-21	pinacotecas	1	0	0	1	0.000	0.000	1.0
TEST-22	catalunya	1	0	0	1	0.000	0.000	1.0
TEST-23	albarracín	1	0	0	1	0.000	0.000	1.0
TEST-24	capiteles	2	0	0	2	0.000	0.000	1.0
TEST-25	salamanca	3	3	2	0	0.933	0.00031	0.000
TEST-26	jerusalén	2	0	0	2	0.000	0.000	1.0
TEST-27	acueducto	1	0	0	1	0.000	0.000	1.0
TEST-28	isabelino	2	2	3	0	0.850	0.00047	0.000
TEST-29	agricultura	1	0	1	1	-0.100	0.00016	1.0
TEST-30	velázquez	2	0	2	2	-0.100	0.00031	1.0
Totales / Occ-weighted Value		81	17	49	64	0.149	0.00776	0.790
Valores medios		2	0	1	2	0.095	0.00026	0.805
ATWV		**-0.060**						

C5. Palabras largas umbral score = -225

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	universidad	9	3	0	6	0.333	0.000	0.667
TEST-02	renacentistas	2	0	2	2	-0.1	0.00031	1.000
TEST-03	arquitectura	15	1	0	14	0.067	0.000	0.933
TEST-04	restaurantes	1	0	0	1	0.000	0.000	1.000
TEST-05	ayuntamiento	4	1	0	3	0.250	0.000	0.750
TEST-06	barcelona	4	2	2	2	0.450	0.00031	0.500
TEST-07	modernismo	2	2	1	0	0.950	0.00016	0.000
TEST-08	plateresca	3	0	0	3	0.000	0.000	1.000
TEST-09	compostela	7	0	1	7	-0.014	0.00016	1.000
TEST-10	hospitales	1	0	0	1	0.000	0.000	1.000
TEST-11	guadalajara	2	0	0	2	0.000	0.000	1.000
TEST-12	cristianos	5	1	0	4	0.200	0.000	0.800
TEST-13	delirante	1	0	3	1	-0.300	0.00047	1.000
TEST-14	guadarrama	2	1	2	1	0.400	0.00031	0.500
TEST-15	extremadura	2	0	0	2	0.000	0.000	1.000
TEST-16	cantábrico	1	0	1	1	-0.100	0.00016	1.000
TEST-17	mudéjares	1	0	0	1	0.000	0.000	1.000
TEST-18	mediterráneo	1	1	1	0	0.900	0.00016	0.000
TEST-19	tenerife	1	0	0	1	0.000	0.000	1.000
TEST-20	cervantino	1	0	0	1	0.000	0.000	1.000
TEST-21	pinacotecas	1	0	0	1	0.000	0.000	1.000
TEST-22	catalunya	1	0	0	1	0.000	0.000	1.000
TEST-23	albarracín	1	0	0	1	0.000	0.000	1.000
TEST-24	capiteles	2	0	0	2	0.000	0.000	1.000
TEST-25	salamanca	3	2	0	1	0.667	0.000	0.333
TEST-26	jerusalén	2	0	0	2	0.000	0.000	1.000
TEST-27	acueducto	1	0	0	1	0.000	0.000	1.000
TEST-28	isabelino	2	2	1	0	0.950	0.00016	0.000
TEST-29	agricultura	1	0	1	1	-0.100	0.00016	1.000
TEST-30	velázquez	2	0	2	2	-0.100	0.00031	1.000
Totales / Occ-weighted Value		81	16	17	65	0.176	0.00269	0.802
Valores medios		2	0	0	2	0.148	0.00009	0.816
ATWV		**0.095**						

C6. Palabras largas sin plurales (umbral score = -225)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	universidad	9	3	0	6	0.333	0.000	0.667
TEST-02	arquitectura	15	1	0	14	0.067	0.000	0.933
TEST-03	ayuntamiento	4	1	0	3	0.250	0.000	0.750
TEST-04	barcelona	4	2	2	2	0.450	0.00031	0.500
TEST-05	modernismo	2	2	1	0	0.950	0.00016	0.000
TEST-06	plateresca	3	0	0	3	0.000	0.000	1.000
TEST-07	compostela	7	0	1	7	-0.014	0.00016	1.000
TEST-08	guadalajara	2	0	0	2	0.000	0.000	1.000
TEST-09	cristianos	5	1	0	4	0.200	0.000	0.800
TEST-10	delirante	1	0	3	1	-0.300	0.00047	1.000
TEST-11	guadarrama	2	1	2	1	0.400	0.00031	0.500
TEST-12	extremadura	2	0	0	2	0.000	0.000	1.000
TEST-13	cantábrico	1	0	1	1	-0.100	0.00016	1.000
TEST-14	mediterráneo	1	1	1	0	0.900	0.00016	0.000
TEST-15	tenerife	1	0	0	1	0.000	0.000	1.000
TEST-16	cervantino	1	0	0	1	0.000	0.000	1.000
TEST-17	catalunya	1	0	0	1	0.000	0.000	1.000
TEST-18	albarracín	1	0	0	1	0.000	0.000	1.000
TEST-19	salamanca	3	2	0	1	0.667	0.000	0.333
TEST-20	jerusalén	2	0	0	2	0.000	0.000	1.000
TEST-21	acueducto	1	0	0	1	0.000	0.000	1.000
TEST-22	isabelino	2	2	1	0	0.950	0.00016	0.000
TEST-23	agricultura	1	0	1	1	-0.100	0.00016	1.000
TEST-24	velázquez	2	0	2	2	-0.100	0.00031	1.000
Totales / Occ-weighted Value		73	16	15	57	0.199	0.00237	0.781
Valores medios		3	0	0	2	0.190	0.00010	0.770
ATWV		**0.132**						

C7. Palabras cortas sin términos confundibles (umbral score = -80)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	madrid	5	0	1	5	-0.020	0.00016	1.000
TEST-02	córdoba	2	2	0	0	1.000	0.00000	0.000
TEST-03	españa	8	0	0	8	0.000	0.00000	1.000
TEST-04	ávila	4	4	35	0	0.125	0.00547	0.000
TEST-05	baeza	6	2	3	4	0.283	0.00047	0.667
TEST-06	picasso	3	0	0	3	0.000	0.00000	1.000
TEST-07	gaudí	15	7	11	8	0.393	0.00172	0.533
TEST-08	cáceres	7	2	1	5	0.271	0.00016	0.714
TEST-09	santiago	13	3	0	10	0.231	0.00000	0.769
TEST-10	teruel	1	0	0	1	0.000	0.00000	1.000
TEST-11	cuenca	4	1	5	3	0.125	0.00078	0.750
TEST-12	sofía	2	0	2	2	-0.100	0.00031	1.000
TEST-13	civil	3	3	7	0	0.767	0.00109	0.000
TEST-14	astorga	1	0	0	1	0.000	0.00000	1.000
TEST-15	antoni	2	1	10	1	0.000	0.00156	0.500
TEST-16	ibiza	4	1	0	3	0.250	0.00000	0.750
TEST-17	platón	1	0	0	1	0.000	0.00000	1.000
TEST-18	quijote	2	0	0	2	0.000	0.00000	1.000
TEST-19	goya	2	0	2	2	-0.100	0.00031	1.000
TEST-20	joan	1	1	25	0	-1.500	0.00391	0.000
TEST-21	ainsa	1	0	37	1	-3.700	0.00578	1.000
TEST-22	somport	1	0	0	1	0.000	0.00000	1.000
TEST-23	colón	2	0	23	2	-1.150	0.00360	1.000
TEST-24	martín	2	0	9	2	-0.450	0.00141	1.000
TEST-25	segovia	3	0	0	3	0.000	0.00000	1.000
TEST-26	greco	3	0	10	3	-0.333	0.00156	1.000
TEST-27	valencia	1	0	2	1	-0.200	0.00031	1.000
Totales / Occ-weighted Value		99	27	183	72	0.088	0.02905	0.727
Valores medios		3	1	6	2	-0.152	0.00106	0.766
ATWV		**-0.826**						

C8. Palabras cortas usando matriz de confusión (con umbral score -80)

Términos		**Estadísticas**						
ID	Texto	Ref	Corr	FA	Miss	Occ. Value	P(FA)	P(Miss)
TEST-01	madrid	5	0	1	5	-0.020	0.00016	1.000
TEST-02	córdoba	2	2	0	0	1.000	0.00000	0.000
TEST-03	españa	8	0	0	8	0.000	0.00000	1.000
TEST-04	ávila	4	4	33	0	0.175	0.00516	0.000
TEST-05	baeza	6	2	3	4	0.283	0.00047	0.667
TEST-06	picasso	3	0	0	3	0.000	0.00000	1.000
TEST-07	gaudí	15	7	11	8	0.393	0.00172	0.533
TEST-08	cáceres	7	2	1	5	0.271	0.00016	0.714
TEST-09	santiago	13	3	0	10	0.231	0.00000	0.769
TEST-10	león	3	3	283	0	-8.433	0.04425	0.000
TEST-11	teruel	1	0	0	1	0.000	0.00000	1.000
TEST-12	cuenca	4	1	5	3	0.125	0.00078	0.750
TEST-13	sofía	2	0	2	2	-0.100	0.00031	1.000
TEST-14	civil	3	3	6	0	0.800	0.00094	0.000
TEST-15	mérida	2	0	12	2	-0.600	0.00188	1.000
TEST-16	astorga	1	0	0	1	0.000	0.00000	1.000
TEST-17	antoni	2	1	10	1	0.000	0.00156	0.500
TEST-18	ibiza	4	1	0	3	0.250	0.00000	0.750
TEST-19	platón	1	0	0	1	0.000	0.00000	1.000
TEST-20	quijote	2	0	0	2	0.000	0.00000	1.000
TEST-21	goya	2	0	12	2	-0.600	0.00188	1.000
TEST-22	joan	1	1	249	0	-23.900	0.03892	0.000
TEST-23	ainsa	1	0	36	1	-3.600	0.00563	1.000
TEST-24	somport	1	0	0	1	0.000	0.00000	1.000
TEST-25	colón	2	0	23	2	-1.150	0.00360	1.000
TEST-26	martín	2	0	8	2	-0.400	0.00125	1.000
TEST-27	segovia	3	0	0	3	0.000	0.00000	1.000
TEST-28	greco	3	0	8	3	-0.267	0.00125	1.000
TEST-29	úbeda	6	1	51	5	-0.683	0.00798	0.833
TEST-30	valencia	1	0	2	1	-0.200	0.00031	1.000
Totales / Occ-weighted Value		110	31	756	79	-0.406	0.12021	0.718
Valores medios		3	1	25	2	-1.214	0.00394	0.751
ATWV		**-3.690**						

D

Archivos, clases y métodos de LatticeSTD

Latt2Multigram.cpp (main)

DetectionFilter.cpp

C_Detection_virt

IsOverlappedWith, IsCenterOverlappedWith, EnlargeTimeBoundaries, PrintDetectionToMLF, SetOverlapThreshold, GetOverlapThreshold, SetTime, GetFromTime, GetToTime, SetScore, GetScore, SetHistory, GetHistory, SetTimeBoundaries.

C_DetectionGroup_virt

JoinDetectionGroup, AddDetection, ComputerGroupScore, SetPrintType, PrintDetectionToMLF.

C_Detections

SetDetectionAdder, SetGroupScoreComputer, SetPrintType, ClearDetections, AddDetection, AddDetectionGroup, RemoveDetectionGroup, RefilterDetections, ComputeScore, PrintDetectionToMLF.

C_GroupScoreComputer_virt

SetDetections, ComputeGroupScore.

C_GroupScoreComputer_FirstScore

ComputeGroupScore

C_GroupScoreComputer_LogAddCenterOverlappedScore

ComputeGroupScore

C_GroupScoreComputer_ContinuousLogAddOverlappedScore

ComputeGroupScore

C_DetectionAdder_virt

SetDetections, AddDetections.

C_DetectionAdder_NoFilter

AddDetection

C_DetectionAdder_GroupTimeBestScore

AddDetection

C_TermsDetection

SetDetectionAdder, SetGroupScoreComputer, SetPrintType, ClearDetections, RefilterDetection, AddTermDetection, ComputeScore, PrintTermDetectionsToMLF.

Latt2MGram.cpp

C_FindMultiGramInLattice

PrepareActualLattice, SetScoringMethod, Search_ExactMatchRecursive, Search_ExactMatch, Search_SubInsDelRecursive, Search_SubInsDel.

argvparser.cpp

ArgvParser

optionKey, isDefinedOption, foundOption, optionValue, parse, arguments, argument, allArguments, usageDescription, errorOption, parseErrorDescription, defineOption, defineOptionAlternative, splitString, setHelpOption, addErrorCode, SetIntroductoryDescription, getAllOptionAlternatives.

ConfMatrix.cpp

C_ConfusionMatrix

PrintVersion, GetSubstitutionProbab, GetInsertionProbab, GetDetectionProbab, ReadConfusionMatrixFromFile, ComputeConfusionProb.

PronDictTree.cpp

C_PronunciationDictionaryTree

GetSplitCharacter, SetSplitCharacter, IsItemInDictionary, GetNextItemFromDictionary, ResetIsNextItemFromDictionary, IsNextItemFromDictionary, ResetRecursiveDictionary, IsPhonemeInRecursiveDictionary, AreItemsInRecursiveDictionary, GetNextItemFromRecursiveDictionary, IsNextItemFromRecursiveDictionary, InsertItemToDictionary, ReadMGramDictionaryFromFile.

C_TreeNode

PronDictTreeTokenPass.cpp

C_PronunciationDictionaryTreeTokenPass

GetSplitCharacter, SetSplitCharacter, SetConfusionMatrix, SetActiveTokenArray, SetFinishedTokenArray, SetConfusionCoefficients, PrintVersion, ActualizeMaxDepth, DeleteAllTokens, DeleteNewActiveTokens, DeleteFinishedTokens, DeleteActiveTokens, Init, MoveNewActiveToActiveTokens, ProcessUnit, ProcessUnitString, GetListOfFoundLabels, RefreshActiveBackPointers, RefreshFinishedBackPointers, CopyTokensArray, IsItemInDictionary, GetNextItemFromDictionary, IsNextItemFromDictionary, ResetIsNextItemFromDictionary, ResetRecursiveDictionary, IsPhonemeInRecursiveDictionary, AreItemsInRecursiveDictionary, GetNextItemFromRecursiveDictionary, IsNextItemFromRecursiveDictionary, InsertItemToDictionary, ReadMGramDictionaryFromFile.

C_TreeNode

AcceptTokenNewActive, AcceptTokenFinished.

C_Token

NextIns, NextSub, NextDel, NextLevel, ShouldTokenDieActual, ShouldTokenDieGlobal, CompeteToken, SetLikelihood, GetLikelihood, AddLikelihood.

E

Presupuesto

Presupuesto

1)	**Ejecución Material**	
	▪ Compra de ordenador personal (Software incluido)	1200 €
	▪ Alquiler de impresora láser durante 10 meses	200 €
	▪ Material de oficina	150 €
	▪ **Total de Ejecución Material**	**1550 €**
2)	**Gastos generales**	
	▪ **sobre los gastos de Ejecución Material**	**248 €**
3)	**Beneficio Industrial**	
	▪ **sobre los gastos de Ejecución Material**	**93 €**
4)	**Honorarios**	
	▪ **900 horas a 15 €/ hora**	**13500 €**
5)	**Material fungible**	
	▪ Gastos de impresión	180 €
	▪ Encuadernación	120 €
	▪ **Total material fungible**	**300 €**
6)	**Subtotal del presupuesto**	
	▪ **Subtotal del Presupuesto**	**15691 €**
7)	**I.V.A. aplicable**	
	▪ **18 % Subtotal del Presupuesto**	**2824,38 €**
8)	**Total del presupuesto**	
	▪ **Total del Presupuesto**	**18.515,38 €**

Madrid, Febrero 2012

El Ingeniero Jefe de Proyecto

Fdo.: Pablo Martín Gila

Ingeniero Superior de Telecomunicación

F

Pliego de condiciones

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un sistema de *DETECCIÓN DE TÉRMINOS ORALES PARA RECUPERACIÓN DE INFORMACIÓN MULTIMEDIA Y SU APLICACIÓN A VÍDEOS DE INFORMACIÓN TURÍSTICA*. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales.

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4 % del presupuesto y la provisional del 2 %.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contratаz anteriormente llamado "Presupuesto de Ejecución Material"que hoy designa otro concepto.

Condiciones particulares.

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.